Interpretability of Deep Learning Systems

Analyzing Interpretability of Deep RCQA Systems & Dialog-Based Image Retrieval

Presented by : Sahana Ramnath
DDP Advisor : Dr. Mitesh M. Khapra (IITM)

June 15, 2020

IIT Madras

Introduction

Project Statements

MOTIVATION

$\Rightarrow$ Explosion of artificial intelligence systems and tasks

$\Rightarrow$ Highly complex neural network architectures approaching human-level accuracy

$\Rightarrow$ Usage of deep models in finance, health, criminal justice etc.

$\Rightarrow$ Need for **trustable** and **interpretable** systems

Two projects as a part of this : *Analyzing Interpretability of Deep RCQA Systems* and *Implementing an Explainable Dialog Based Image Retrieval System*.

Analyzing Interpretability of RCQA Systems

Introduction and Preliminaries

Introduction

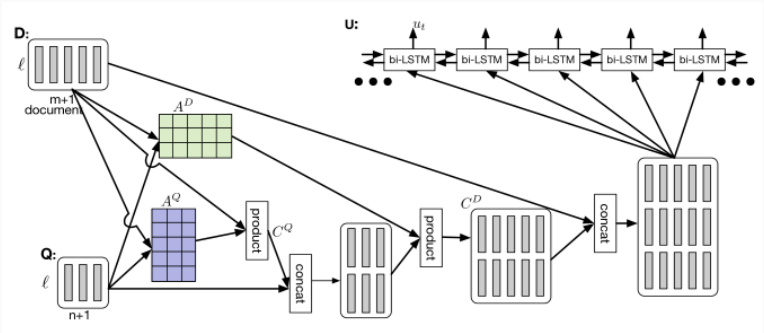
Dataset	Human Performance	SOTA
SQuAD	89.45%	92.42%
RACE	94.5%	89.4%
WikiHop	85.0%	78.3%
HotpotQA	91.4%	82.19%

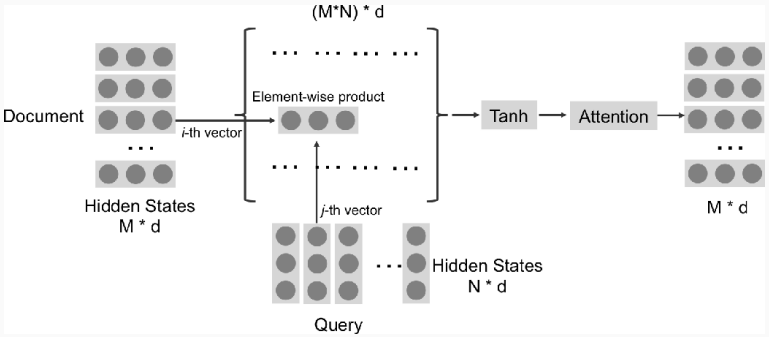
$\Rightarrow$ Deep models approaching human-level performance in RCQA

Introduction

Dataset	Human Performance	SOTA
SQuAD	89.45%	92.42%
RACE	94.5%	89.4%
WikiHop	85.0%	78.3%
HotpotQA	91.4%	82.19%

$\Rightarrow$ Deep models approaching human-level performance in RCQA

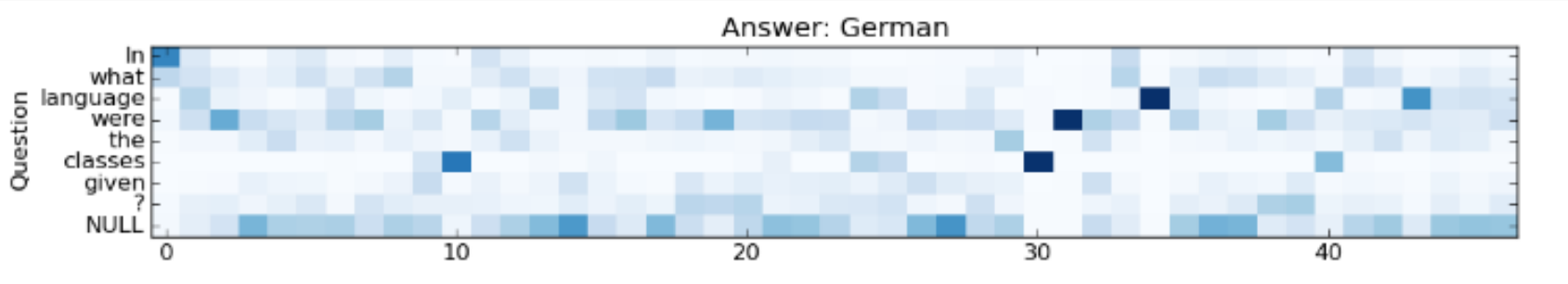




$\Rightarrow$ New complex modules designed for specific functions, eg:- co-attention, self-attention

Analyzing Attention Mechanisms

⇒ Traditional way of analyzing model's predictions - visualizing attention scores for a few examples



⇒ Need to analyze all the model's layers, not just attention!

⇒ Contemporary works : more thorough analysis of attention mechanisms using statistical measures

⇒ Using various gradient-based attribution methods

Preliminaries : Dataset

There exist many datasets today for the task of RCQA. In this project, we use **SQuAD**.

Passage: the panthers finished the regular season with a 15 – 1 record, ... the broncos ... finished the regular season with a 12 – 4 record. They joined the patriots , dallas cowboys , and pittsburgh steelers as one of **four** teams that have made eight appearances in the super bowl.
Question: **How many teams have been in the super bowl eight times?**

Table 1: Example from SQuAD. **Blue** shows the answer, purple shows the contextual passage words and **green** shows the query

.

Preliminaries : QA Framework and Models

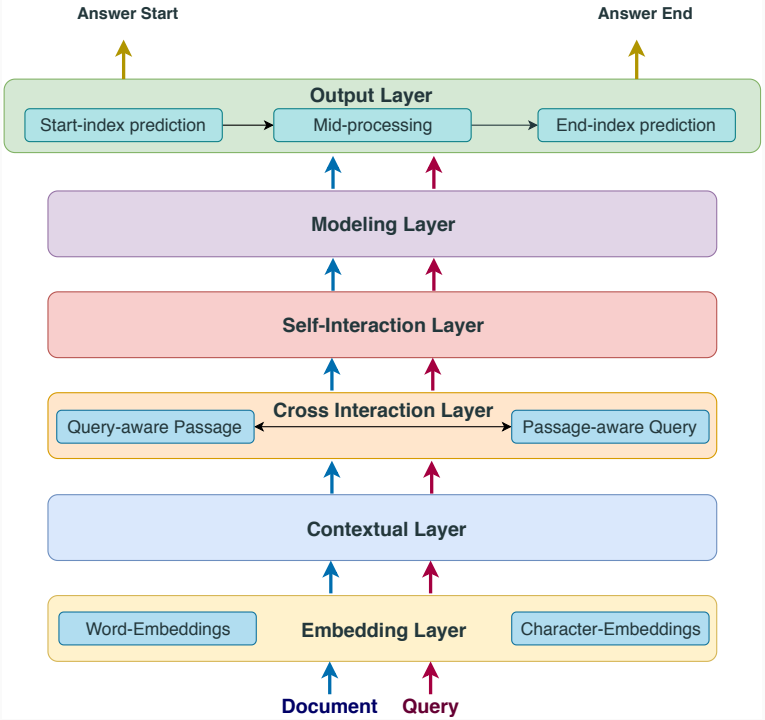


Figure 1: Basic Framework for RCQA

- **Word Embedding Layer**
- **Contextual Layer**
- **Cross-Interaction Layer**
- **Self-Interaction Layer**
- **Modeling Layer**
- **Output Layer**

In this work, we analyze four published models - **BERT**, **BiDAF**, **DCN** and **QANet**.

Preliminaries : Integrated Gradients

Many techniques to attribute a deep network's predictions to its input features - LIME, **Integrated Gradients**, DeepLift, LRP,etc.

DEFINITION : The Integrated Gradient for a passage word w_i, embedded as $x_i \in \mathbf{R}^L$ is computed as follows:

$$IG(x_i) = \int_{\alpha=0}^{1} \frac{\partial M(\tilde{x} + \alpha(x_i - \tilde{x}))}{\partial x_i} \; d\alpha$$

where $\tilde{x}$ is a zero vector, that serves as a baseline to measure integrated gradient for w_i. In this work, the above integral is approximated across 50 uniform samples between $[0, 1]$.

Paradigms of Interpretability

In this work, we broadly analyze the four QA models on the following three questions :

- Can they generate a meaningful rationale to explain their predictions?
- Do they progressively learn more useful/refined word representations at each layer?
- Do specialised layers such as the query-document interaction layer, indeed serve their intended purpose?

We further perform an independent analysis on BERT to understand the QA-specific role performed by each of its layers.

Methods : QA Task Description

For a given passage D with d words $[w_1, w_2, \ldots, w_d]$, question Q, answer $[i, j]$, where i and j are start and end indices of answer span in D and a model M with ϕ parameters, the answer prediction task is modeled as:

$$\boxed{p(w_s, w_e) = M(w_s, w_e | \text{embed}(D), \text{embed}(Q), \phi)}$$

w_s, w_e are the predicted answer span's start and end indices. The passage and question words are embedded using embed(.), that encodes a word to $\mathbf{R}^L$ space.

Methods : QA Task Description

For any given layer l, the above is equivalent to:

$$\boxed{p(w_s, w_e) = f_l(w_s, w_e | E_{l-1}(P), E_{l-1}(Q), \theta)}$$

where f_l is the forward propagation from layer l to the prediction. $E_l(.)$, is the representation learnt for passage or query words by a given layer l. To elaborate, we consider the network below the layer l as a blackbox which generates *input* representations for layer l.

Methods : Layerwise Integrated Gradients

Algorithm 1 To compute Layer-wise Integrated Gradients for layer l

1: $\tilde{p} = 0$ //zero baseline
2: $m = 50$
3: $G_l(p) = \frac{1}{m} \sum_{k=1}^{m} \frac{\partial f_l(\tilde{p} + \frac{k}{m}(p - \tilde{p}))}{\partial E_l}$
4: $IG_l(p) = [(p - \tilde{p}) \times G_l(p)]$
5: // Compute squared norm at each row
6: $\tilde{I}_l([w_1, \ldots, w_k]) = ||IG_l(p)||$
7: Normalize $\tilde{I}_l$ to a probability distribution I_l

Analysis on Extracted Rationale

Motivation

Some existing works

- Test models with adversarial datasets [11, 5]
- Introduce models with rationale-generating components [7]

Issues

- Creating the right adversarial dataset is difficult
- Models with self-explaining components *cannot* be used to analyze other models

Motivation

Some existing works

- Test models with adversarial datasets [11, 5]
- Introduce models with rationale-generating components [7]

Issues

- Creating the right adversarial dataset is difficult
- Models with self-explaining components *cannot* be used to analyze other models

Hence in this section, we use a simple extensible framework to extract a QA model's rationale using integrated gradients.

Model's Rationale

A model's rationale is the minimal set of items which when removed collectively, causes a change in the model's prediction. [1]

[1] adopted from [9]

Model's Rationale

A model's rationale is the minimal set of items which when removed collectively, causes a change in the model's prediction. [2]

Algorithm 2 Rationale Extraction

$\tilde{D}([w_1, ..., w_d]) \leftarrow ||IG(w_1)||, ..., ||IG(w_d)||$
// normalize for a probability distribution
$\tilde{D} \leftarrow \tilde{D}/\text{sum}(\tilde{D})$
$\mathbf{X} \leftarrow$ Rank $[w_1, \ldots, w_d]$ based on $\tilde{D}$
rationale = [ ]
repeat
    rationale.insert(pop($\mathbf{X}$))
until (Decision Flips)

[2] adopted from [9]

Example : Model's Rationale

Question: What was maria curie the first female recipient of ?	
Answer: Nobel Prize	
DCN	One of the most famous people born in Warsaw was Maria SkÃÂodowska-Curie , who achieved international recognition for her research on radioactivity and was the first female recipient of the Nobel Prize....
BiDAF	One of the most famous people born in Warsaw was Maria SkÃÂodowska-Curie , who achieved international recognition for her research on radioactivity and was the first female recipient of the Nobel Prize....
QANET	One of the most famous people born in Warsaw was Maria SkÃÂodowska-Curie , who achieved international recognition for her research on radioactivity and was first female recipient of the Nobel Prize....
BERT	One of the most famous people born in Warsaw was Maria SkÃÂodowska-Curie , who achieved international recognition for her research on radioactivity and was the first female recipient of the Nobel Prize....

Table 2: Sample of Rationale Extracted for DCN, BiDAF, QANET and BERT. All the models predicted the correct answer "Nobel Prize".

.

Quantitative Analysis on Rationale Extracted

Hence, a model is more explainable if a smaller set of words causes a decision-flip, i.e., rationale is more concise.

$$\boxed{\textit{flip\_fraction} = \frac{\textit{card}(\textit{rationale})}{\textit{card}(\textit{passage})}}$$

The flip-fraction is the fraction of words in the passage that are part of the rationale.

By the given definition, lower the flip-fraction, more explainable the model is.

Quantitative Analysis on Rationale Extracted

Model	Mean	Variance
BERT	0.072	0.03
BiDAF	0.161	0.064
QANet	0.165	0.065
DCN	0.215	0.059

Table 3: Mean and Variance of flip-fraction on SQuAD's dev set, scale of 0-1

Observation : Low flip-fraction values observed for all models.

Conclusion : Model is able to provide correct attribution to input items (able to highlight words important to its prediction) .

Quantitative Analysis on Rationale Extracted

Conclusion : Model is able to provide correct attribution to input items.

Limitation : Lower flip-fraction doesn't ensure quality of rationale.
Model could be selecting :

$\Rightarrow$ wrong words

$\Rightarrow$ insufficient explanation

$\Rightarrow$ answer span itself

Human Evaluations on Rationale Extracted

For 500 randomly selected examples, **human rationale** is collected.

With this as baseline, the model's rationale is evaluated for **precision** and **recall (completeness)**.

Human Evaluations on Rationale Extracted

For 500 randomly selected examples, **human rationale** is collected.

With this as baseline, the model's rationale is evaluated for **precision** and **recall (completeness)**.

The model is defined as **precise** if every word in its rationale is also present in the human-annotated rationale.

It is defined as **complete** if every word in the human rationale is present in its rationale.

Human Evaluations on Rationale Extracted

Model	Incl. Answer Span			Excl. Answer Span		
	% P	% R	% F1	% P	% R	% F1
BERT	95.2	17.2	29.1	84.4	6.25	11.6
BiDAF	85.8	19.8	32.2	66.9	12.6	21.2
DCN	65.1	26.9	38.1	42.5	18.6	25.9
QANet	83.1	19.6	31.7	62.6	11.1	18.9

Table 4: Precision(P), Recall(R) and F1 score of overlap of model's rationale with human rationale

Observation : High precision in both cases, however, 10-20% drop when answer span is removed.

Conclusion : Models highlight correct words, however, they are highly focused on the answer span.

Human Evaluations on Rationale Extracted

Model	Incl. Answer Span			Excl. Answer Span		
	% P	% R	% F1	% P	% R	% F1
BERT	95.2	17.2	29.1	84.4	6.25	11.6
BiDAF	85.8	19.8	32.2	66.9	12.6	21.2
DCN	65.1	26.9	38.1	42.5	18.6	25.9
QANet	83.1	19.6	31.7	62.6	11.1	18.9

Table 5: Precision(P), Recall(R) and F1 score of overlap of model's rationale with human rationale

Observation : Recall values are much lower.

Conclusion : Hence, though models seem explainable based on flip-fraction/precision, their rationale is highly incomplete. Reasoning on how the answer was retrieved is not yet sufficient.

Conicity : Evolution of Embeddings

Conicity

Motivation : To analyze the evolution of word embeddings across the model's layers.

Metric : Conicity[10] is used to calculate the aggregated vector space similarity of these embeddings.

Conicity

Motivation : To analyze the evolution of word embeddings across the model's layers.

Metric : Conicity[10] is used to calculate the aggregated vector space similarity of these embeddings.

Conicity of a set of vectors $\mathbf{V} = \{\mathbf{v}_1, \ldots, \mathbf{v}_m\}$ is defined as follows :

$$\text{ATM}(\mathbf{v}_i, \mathbf{V}) = \text{cosine}(\mathbf{v}_i, \frac{1}{m}\sum_{j=1}^{m}\mathbf{v}_j) \text{ (alignment to mean)}$$

$$\text{Conicity}(\mathbf{V}) = \frac{1}{m}\sum_{i=1}^{m}\text{ATM}(\mathbf{v}_i, \mathbf{V}) \text{ (mean of all ATM's)}$$

Conicity

Motivation : To analyze the evolution of word embeddings across the model's layers.

Metric : Conicity[10] is used to calculate the aggregated vector space similarity of these embeddings.

Conicity of a set of vectors $\mathbf{V} = \{\mathbf{v}_1, \ldots, \mathbf{v}_m\}$ is defined as follows :

$$\text{ATM}(\mathbf{v}_i, \mathbf{V}) = \text{cosine}(\mathbf{v}_i, \frac{1}{m}\sum_{j=1}^{m}\mathbf{v}_j) \text{ (alignment to mean)}$$

$$\text{Conicity}(\mathbf{V}) = \frac{1}{m}\sum_{i=1}^{m}\text{ATM}(\mathbf{v}_i, \mathbf{V}) \text{ (mean of all ATM's)}$$

A high value of conicity means that the vectors in $\mathbf{V}$ form a narrow cone, centered at the origin; that is, they are all highly aligned with each other.

Conicity at Each Layer

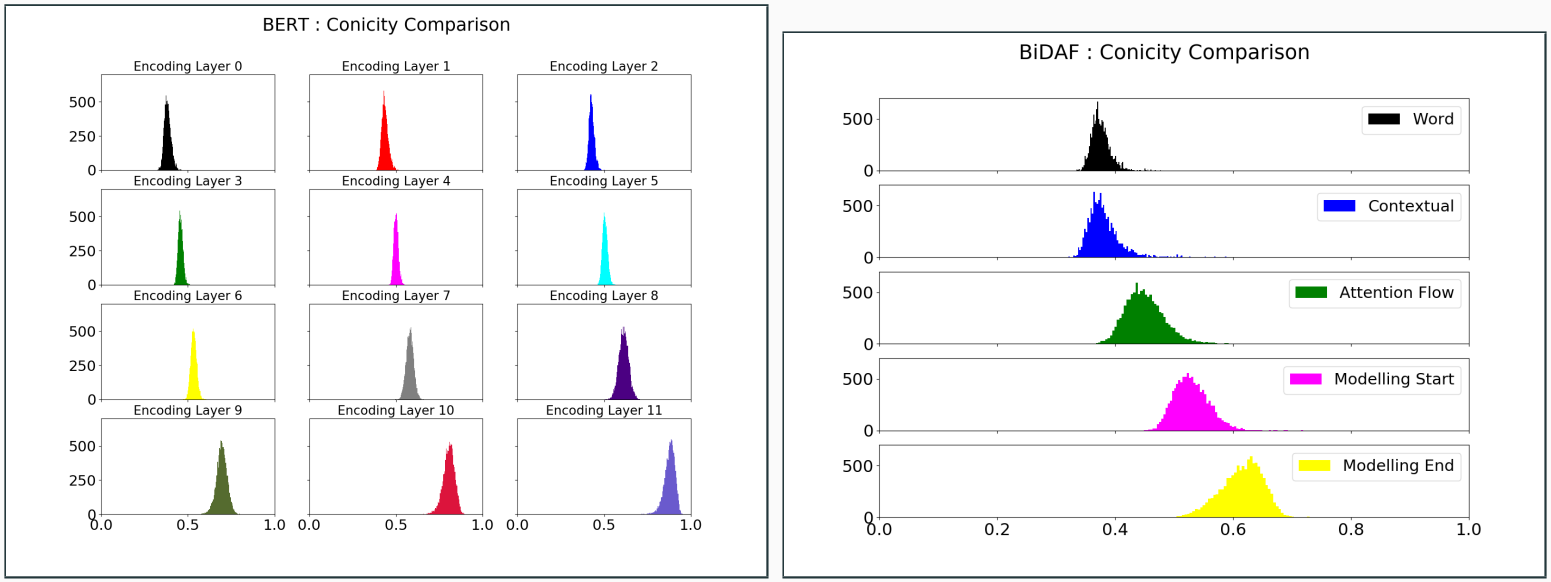


Figure 2: Histogram of conicity across layers for BERT(left to right; top to bottom) and BiDAF(top to bottom)

Observation : Conicity increases from the initial layers to the deeper layers.

Conclusion : Later layers have **lesser discriminatory powers**.

Conicity at Each Layer

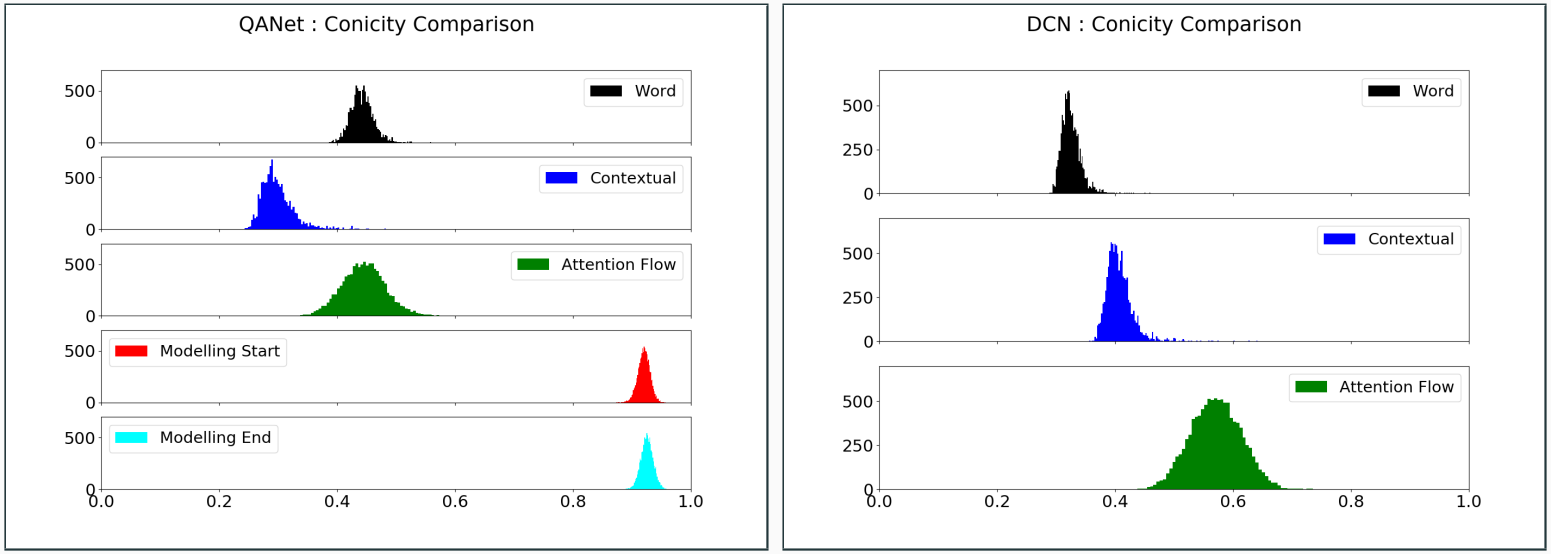


Figure 3: Histogram of conicity across layers for QANET(top to bottom) and DCN(top to bottom)

Limitation : This graph doesn't show the conicity for answer span words and other words separately.

Could be that all the words' embeddings come together, but the answer span words alone are farther away and hence distinguishable.

Conicity Across Layers : BERT

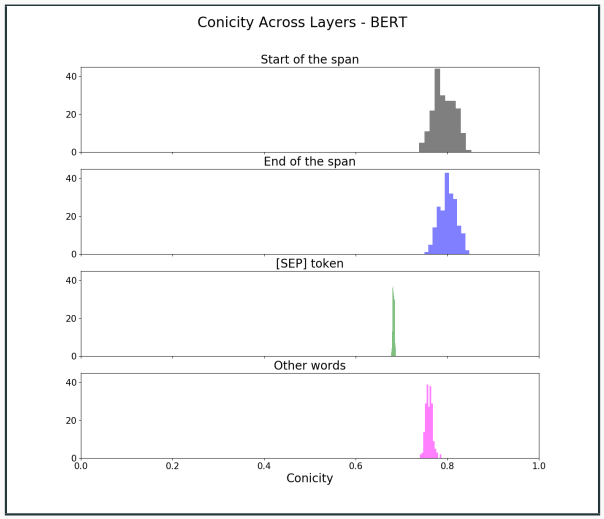


Figure 4: Conicity - Evolution of word embeddings across layers in BERT

Observation : Conicity is fairly high.

Conclusions :

(i) Embeddings don't vary much across layers.

(ii) These small mathematical perturbations in embeddings across layers are significant to the model's working; current methods like conicity are not sensitive enough to capture such changes.

Interpreting BERT's Layers

Motivation

- BERT has replaced SOTA in multiple NLP tasks.
- Earlier works [1, 12, 8] allocate syntactic and semantic purposes for each layer in BERT for simple tasks : sentiment classification, NLI, etc.
- Not many works in analyzing BERT for **RCQA**.

Motivation

- BERT has replaced SOTA in multiple NLP tasks.
- Earlier works [1, 12, 8] allocate syntactic and semantic purposes for each layer in BERT for simple tasks : sentiment classification, NLI, etc.
- Not many works in analyzing BERT for **RCQA**.

Analyzing BERT for QA is challenging because :

(i) BERT's sheer non-linearity and number of parameters

(ii) BERT does not have pre-defined roles across layers as compared to pre-BERT models like BiDAF, DCN, etc.

Motivation

- BERT has replaced SOTA in multiple NLP tasks.
- Earlier works [1, 12, 8] allocate syntactic and semantic purposes for each layer in BERT for simple tasks : sentiment classification, NLI, etc.
- Not many works in analyzing BERT for **RCQA**.

Analyzing BERT for QA is challenging because :

(i) BERT's sheer non-linearity and number of parameters

(ii) BERT does not have pre-defined roles across layers as compared to pre-BERT models like BiDAF, DCN, etc.

In this section, various experiments are performed to try to map BERT's layers to the QA-specific functions.

Layer Functionality : Definition

$\Rightarrow$ A layer's functionality is defined as how it distributes importance over the passage words, using the distribution I_l.

$\Rightarrow$ To compute the similarity between any two layers x, y, **Jensen-Shannon Divergence (JSD)** is measured between their corresponding importance distributions I_x, I_y.

$\Rightarrow$ The JSD scores are calculated between every pair of layers in the model, and are visualised as a $n_l \times n_l$ heatmap (n_l is the number of layers).

$\Rightarrow$ Higher JSD score corresponds to the two layers being more different.

Layer Functionality : Example

Question: Why was Polonia relegated from the country's top flight in 2013?
Answer: disastrous financial situation

L0	Polonia was relegated from the country's top flight in 2013 because of their disastrous financial situation. They are now playing in the 4th league....	**L9**	Polonia was relegated from the country's top flight in 2013 because of their disastrous financial situation. They are now playing in the 4th league....
L1	Polonia was relegated from the country's top flight in 2013 because of their disastrous financial situation. They are now playing in the 4th league....	**L10**	Polonia was relegated from the country's top flight in 2013 because of their disastrous financial situation. They are now playing in the 4th league....
L2	Polonia was relegated from the country's top flight in 2013 because of their disastrous financial situation. They are now playing in the 4th league....	**L11**	Polonia was relegated from the country's top flight in 2013 because of their disastrous financial situation. They are now playing in the 4th league....

Table 6: Sample l_l over BERT's first and last 3 layers, visualised as a heatmap

Layer Functionality : Overall JSD

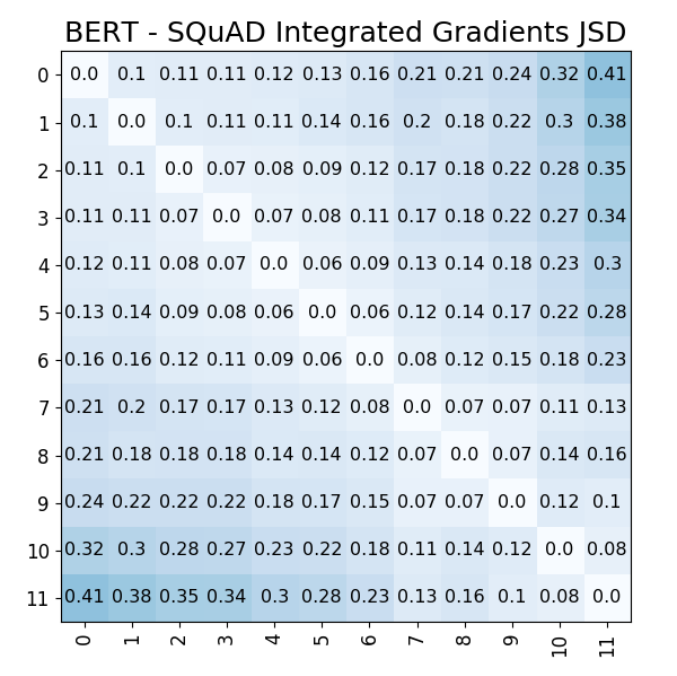


Figure 5: BERT : Pairwise JSD between l_l's

Observation : Low JSD between layers, with only minimal increase as the layers are apart (min/max is just 0.06/0.41).

Conclusion (Preliminary) : The layers are highly similar to each other.

Layer Functionality : JSD with top-k retained/removed

Method : To evaluate the source of similarity, the l_l's are analyzed in 2 ways :

- only head (top-k words) retained, rest zeroed out
- top-k words zeroed out, the remaining words (the tail) retained

In both cases, the distribution is re-normalized to maintain the probability sum.

Layer Functionality : JSD with top-k retained/removed

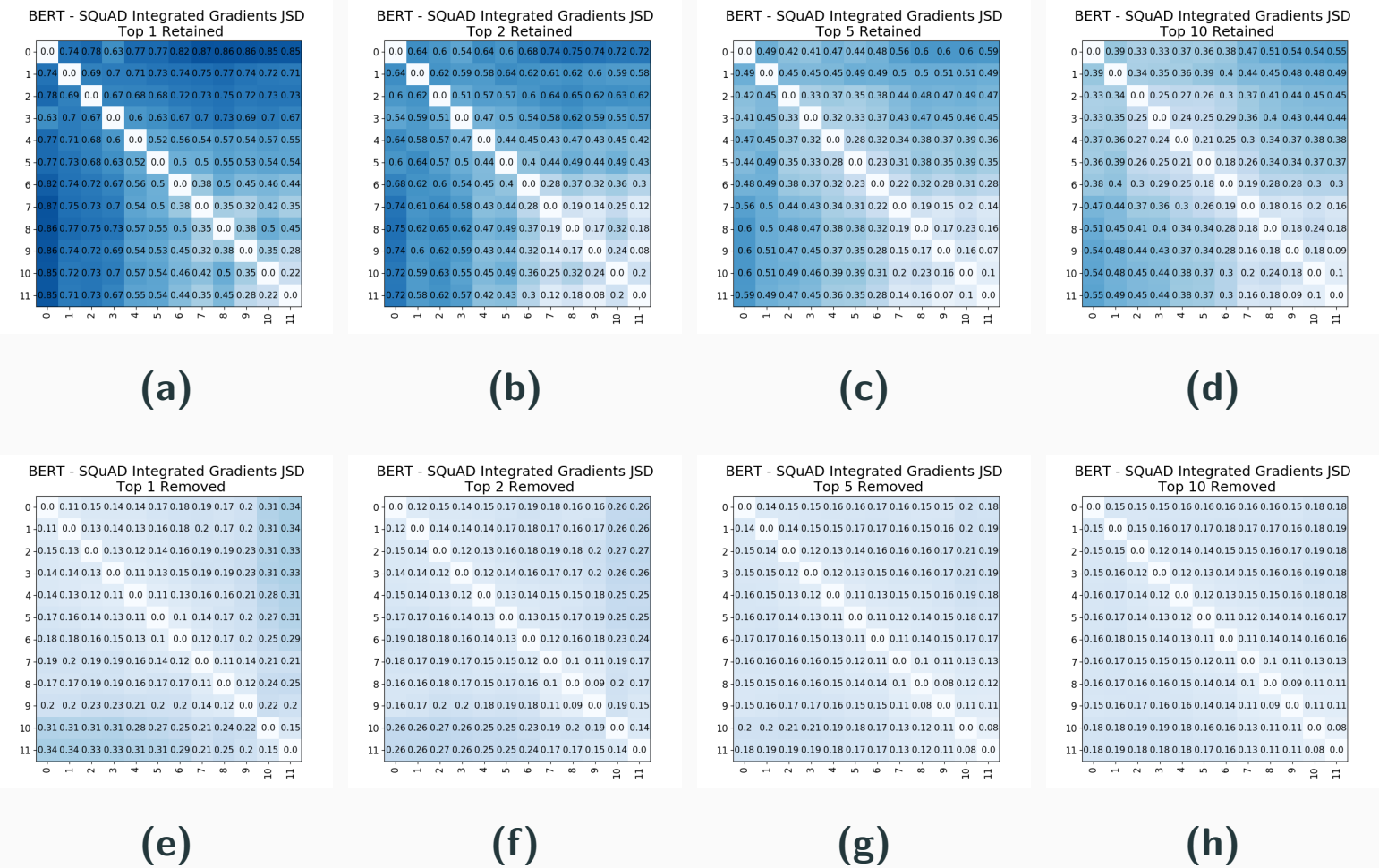


Figure 6: JSD between l_i's with top-k items retained/removed

Layer Functionality : JSD with top-k retained/removed

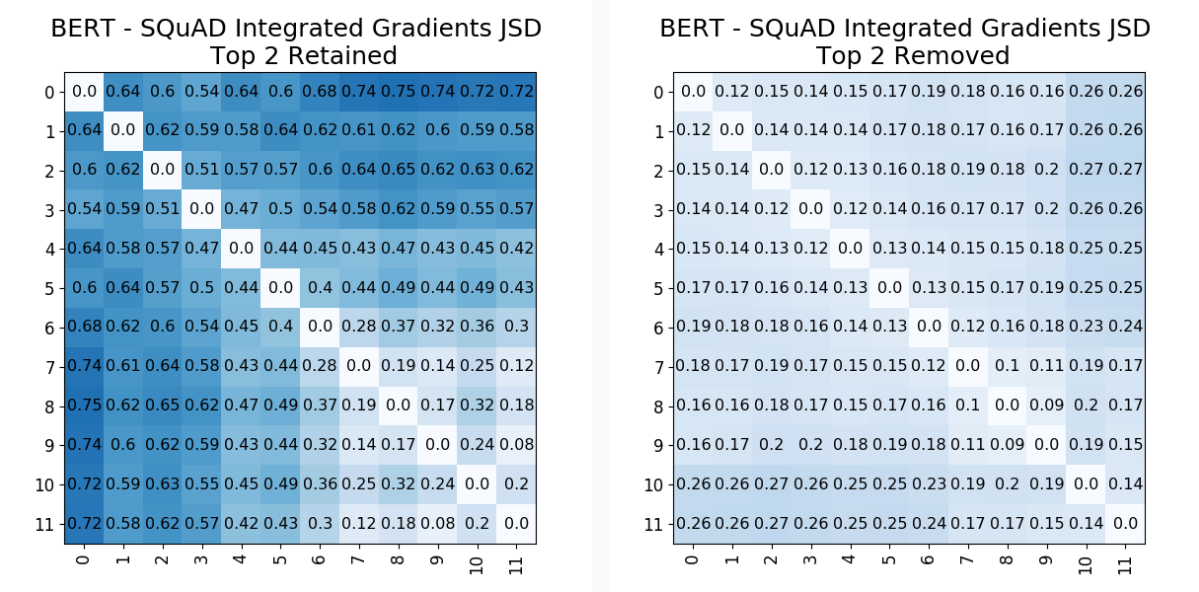


Figure 7: Top-2 words retained/removed

Observations :

(i) Higher values when top-2 words are retained (min 0.08/max 0.72) versus removed (min 0.09/max 0.26).

(ii) For both head and tail, as k increases from 1 to 10, JSD scores decrease.

Layer Functionality : JSD with top-k retained/removed

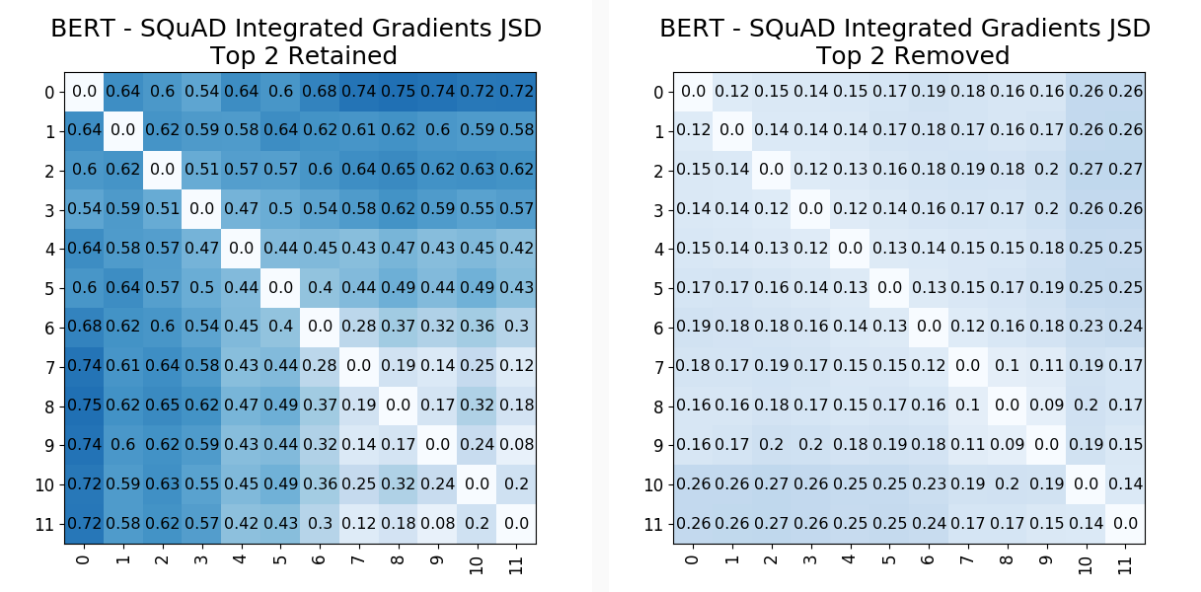


Figure 8: Top-2 words retained/removed

Conclusions :

(i) A layer's functionality is reflected by the head (top-k words).

(ii) As top-k words are removed, there's an almost uniform distribution over the lesser important words.

Layer Functionality : JSD split by question-type

Motivation : The model approaches different question types differently.

For example

(i) “what” or “who” questions require entities as answers, and in SQuAD can probably be answered directly

(ii) questions like “why” or “how” require a more in-depth reading of the passage.

Layer Functionality : JSD split by question-type

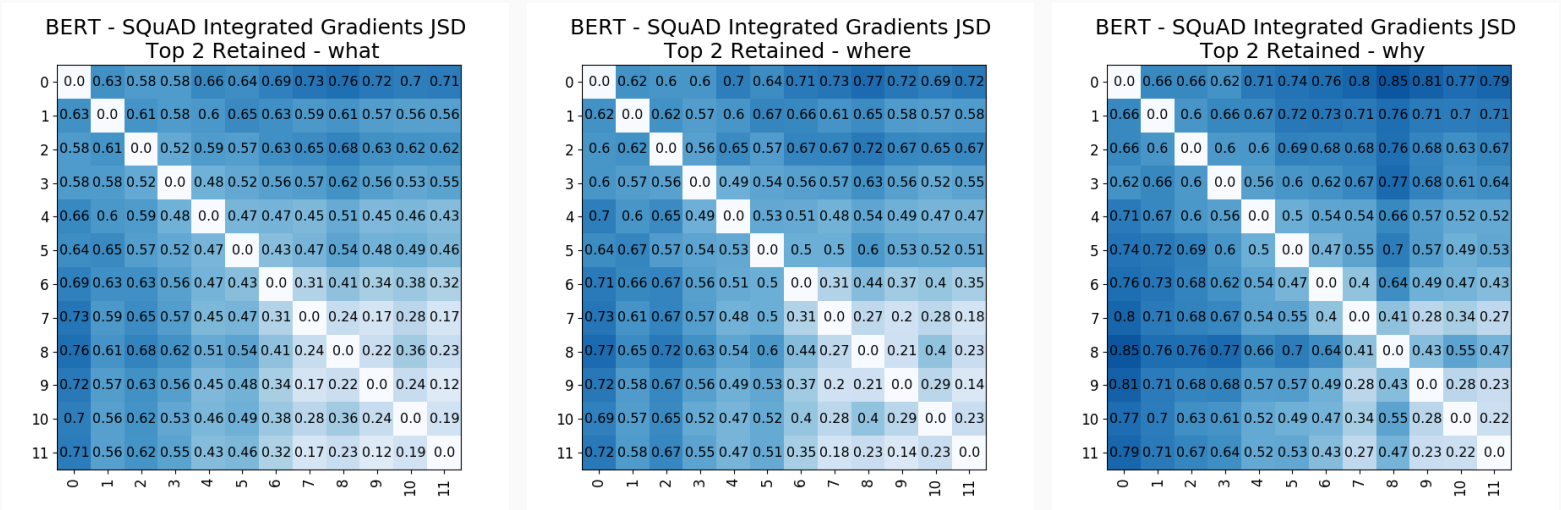


Figure 9: JSD of l_i's, split by question types (top-2 retained)

The "what", "where" heatmaps are similar to previous heatmaps.

However, the "why" heatmap, shows a **slightly higher JSD in the later layers** as well, supporting the hypothesis that such questions require a deeper understanding of the passage and hence more work to be done by the model.

Layer Functionality : Probing for QA roles

Method : Based on their functionality l_l, the layers are analyzed to see which of them focus more on the question, the context around the answer, etc.

The passage words are segregated into three categories: **answer words, supporting words, query words**.

Following the discussion in the JSD experiment, the top-5 words marked as important in l_l are taken to represent the layer l.

Layer Functionality : Probing for QA roles

Layer	% answer span	% Q-words	% Support Words
L0	26.99	22.94	9.45
L1	26.09	24.35	9.43
L2	29.9	22.41	11.65
L3	30.44	19.55	11.13
L4	30.06	18.33	11.23
L5	30.75	14.71	11.57
L6	31.25	15.33	11.94
L7	32.37	12.29	12.32
L8	30.78	18.91	12.07
L9	34.58	10.21	13.41
L10	34.31	10.56	13.39
L11	34.63	12.0	13.74

Table 7: Semantic statistics about top-5 words

Observations :

(i) Focus on query decreases from initial to final layers.

(ii) Focus on passage words remain fairly constant.

(iii) Focus on the (predicted) answer span increases.

Layer Functionality : Probing for QA roles

Layer	% answer span	% Q-words	% Support Words
L0	26.99	22.94	9.45
L1	26.09	24.35	9.43
L2	29.9	22.41	11.65
L3	30.44	19.55	11.13
L4	30.06	18.33	11.23
L5	30.75	14.71	11.57
L6	31.25	15.33	11.94
L7	32.37	12.29	12.32
L8	30.78	18.91	12.07
L9	34.58	10.21	13.41
L10	34.31	10.56	13.39
L11	34.63	12.0	13.74

Table 8: Semantic statistics about top-5 words

Conclusions :

(i) Initial layers focus on connecting the query and passage.

(ii) Later layers focus on enhancing and verifying the model's prediction.

Visualising Word Representations

Method : Qualitative analysis of word embeddings using t-SNE plots.

Passage: the panthers finished the regular season with a 15 – 1 record, ... the broncos ... finished the regular season with a 12 – 4 record. They joined the patriots , dallas cowboys , and pittsburgh steelers as one of teams that have made **eight** appearances in the super bowl .
Question: **How many appearances have the Broncos made in the super bowl?**

Table 9: Sample from the dev-split of SQuAD. **Blue** shows the answer, purple shows the contextual passage words and **green** shows the query

.

Visualising Word Representations

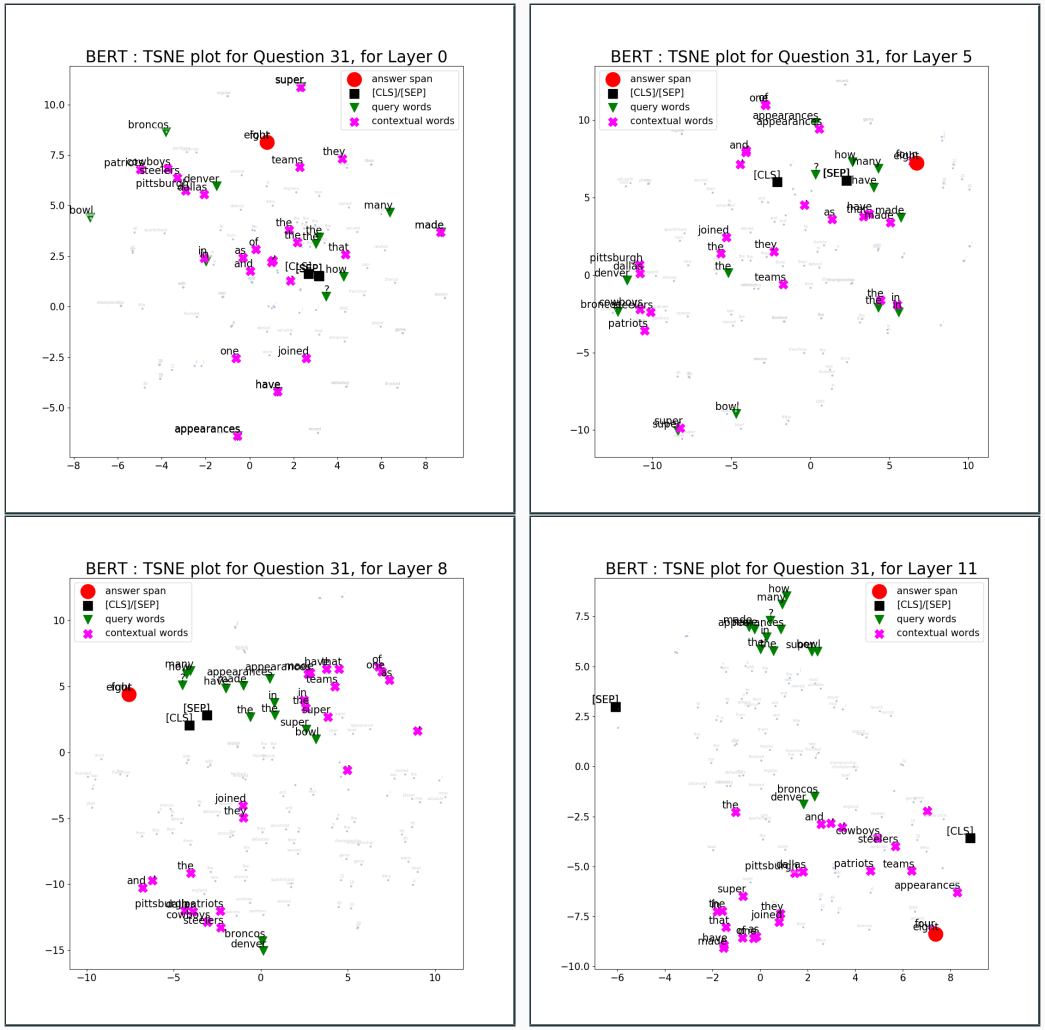


Figure 10: T-sne plots across Layer 0, 5, 8, 11 to analyse how the word representations evolve across layers.

Visualising Word Representations

Observations :

(i) In initial layers (such as layer 0), similar words are closer to each other: such as stop-words, team names, numbers {eight, four} etc.

(ii) From Layer 5 onwards, the passage, question, and answer come close to each other.

(iii) By layer 8, the answer words are segregated from the rest of the words, even though the passage word 'four' which is of the same type as the answer 'eight'(number) is still close to 'eight'.

(iv) In later layers the question words separate from the answer and the supporting words.

(v) Across all 12 layers, embeddings for *four, eight* remain very close together, which could have easily led to the model making a wrong prediction. However, the model still predicts the answer 'eight' correctly.

Quantifier Questions

Experiment : Analyse quantifier questions - **how much/how many**, that could potentially have many confusing answers (numerical quantities - N_q) in the passage.

$$\boxed{ratio = \frac{card(N_q \text{ in top-5})}{card(N_q \text{ in passage})}}$$

This represents the ratio of confusing words marked as important by each layer.

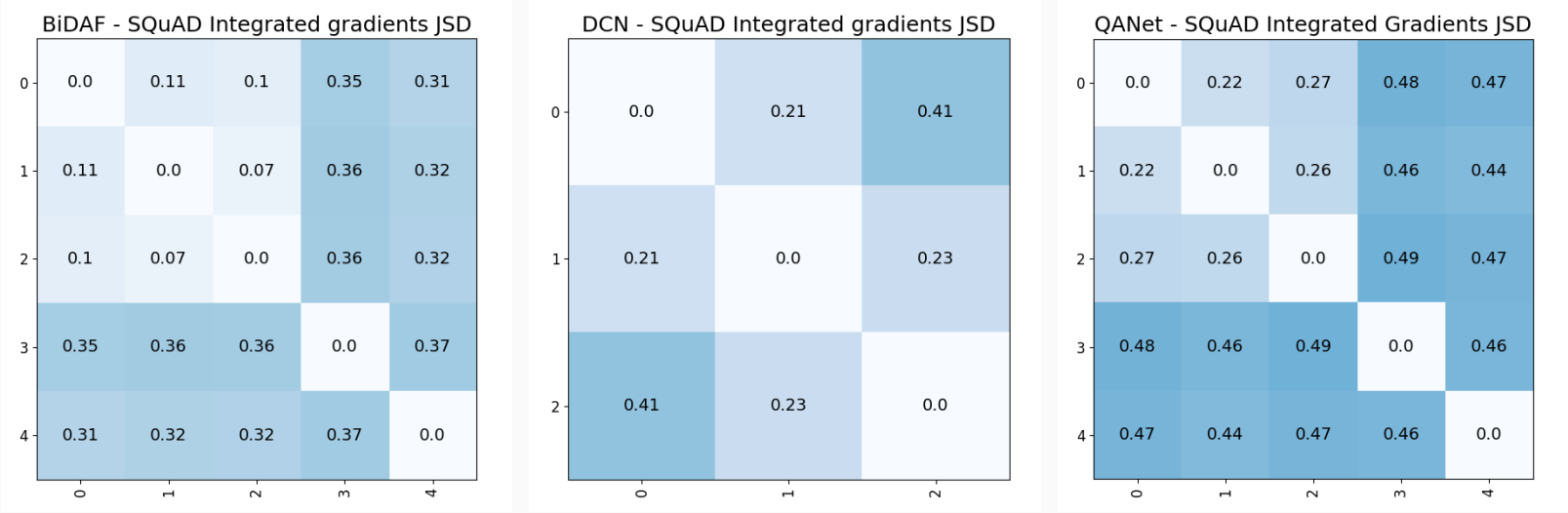
Quantifier Questions

Result : Interestingly, this ratio **increases** from the intial to final layers (5% at layer 0 to 18% layer 11).

This shows that BERT, in its later layers, **distributes its importance over potentially confusing word**s; however, it still manages to predict the correct answer for such questions (87.42% EM for how much/many questions).

This behavior is very different from the assumed roles a layer might take to answer the question; it would have been expected that such words were considered in the initial rather than final layers.

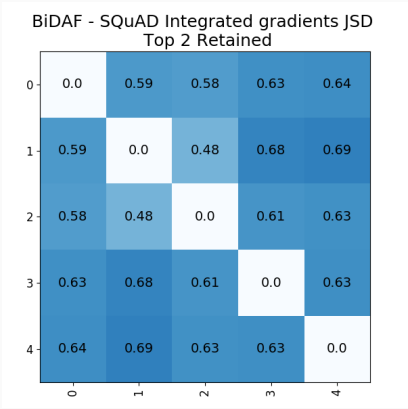
JSD Analysis on BiDAF, DCN, QANET



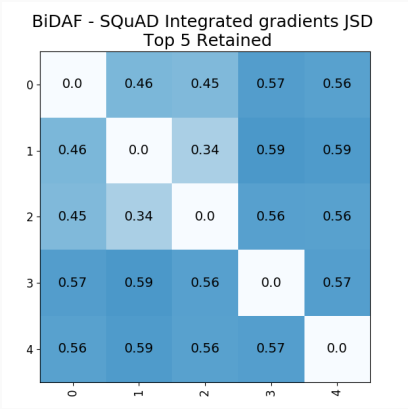
(a) **(b)** **(c)**

Figure 11: Overall JSD for BiDAF, DCN, QANET

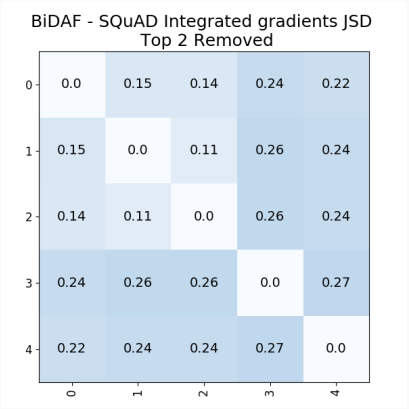
JSD Analysis on BiDAF, DCN, QANET



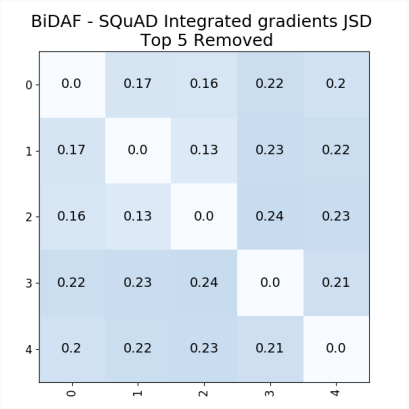
(a)



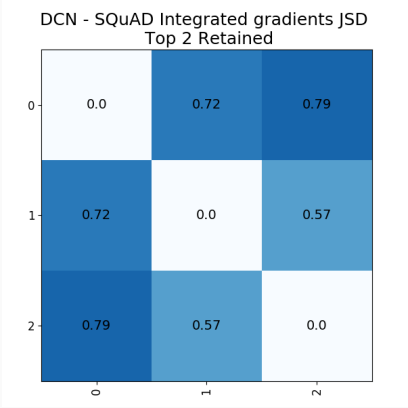
(b)



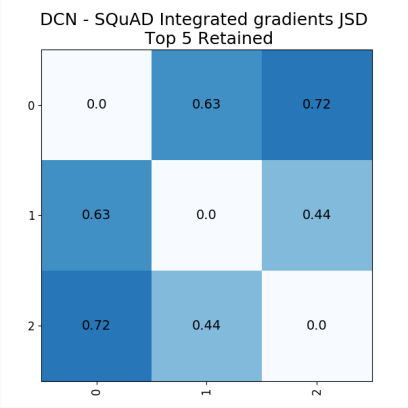
(c)



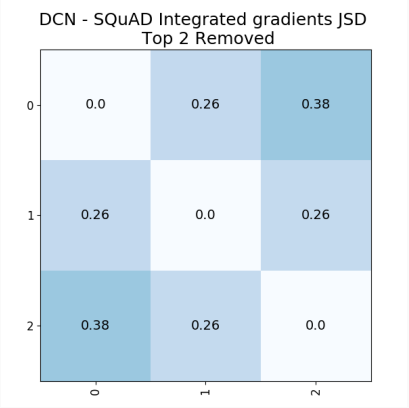
(d)



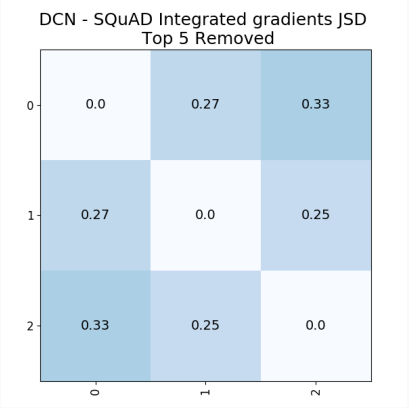
(e)



(f)



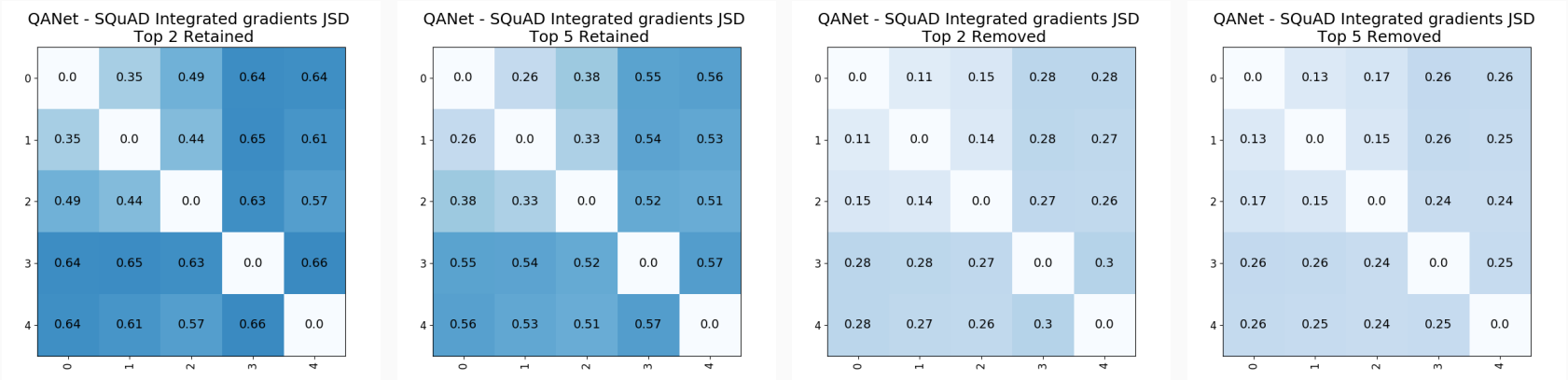
(g)



(h)

Figure 12: JSD of I_l's with top-k retained/removed for BiDAF and DCN

JSD Analysis on BiDAF, DCN, QANET



(a) **(b)** **(c)** **(d)**

Figure 13: JSD of I_l's with top-k retained/removed for QANET

Attention Mechanisms and LIME

Visualisation of Attention Mechanisms

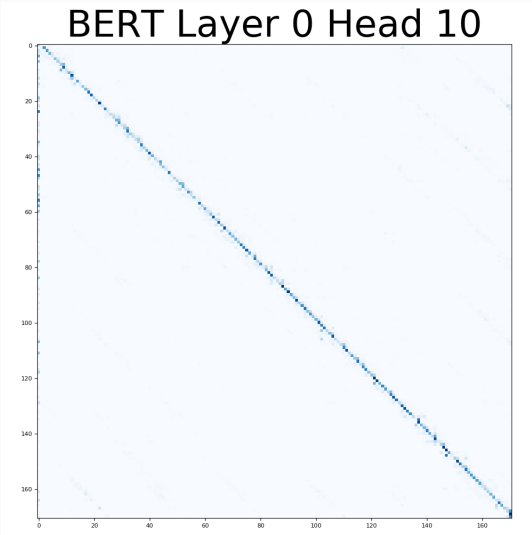
Motivation : Even though attention mechanisms don't explain the model fully, can we still get some information from them?

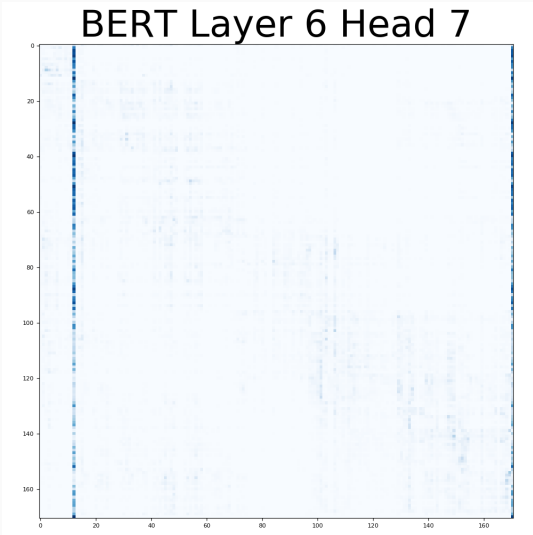
Each transformer layer in BERT implements multi-head self- and co-attention mechanisms, giving 12 attention plots for each layer.

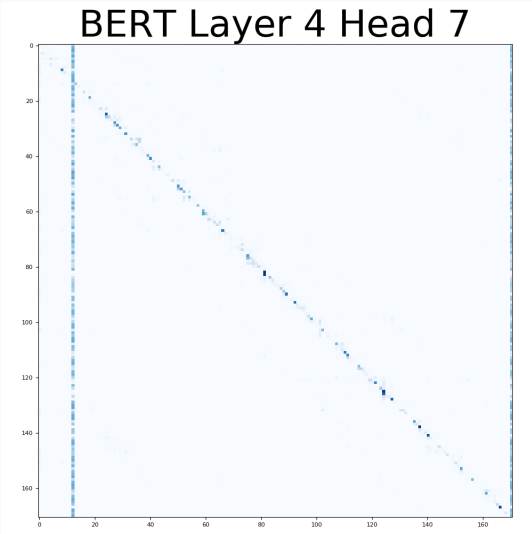
BiDAF, DCN, QANET have passage-query coattention layers.

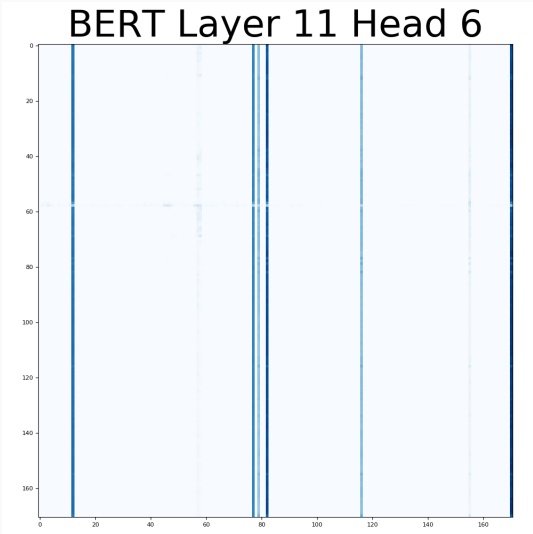
This serves as a purely observational experiment, with an intent to search for any **interesting patterns** in the plots.

Visualisation of Attention Mechanisms









- Positional Heads
- Focus on [CLS] and [SEP] tokens
- Focus on punctuation tokens

Visualisation of Attention Mechanisms

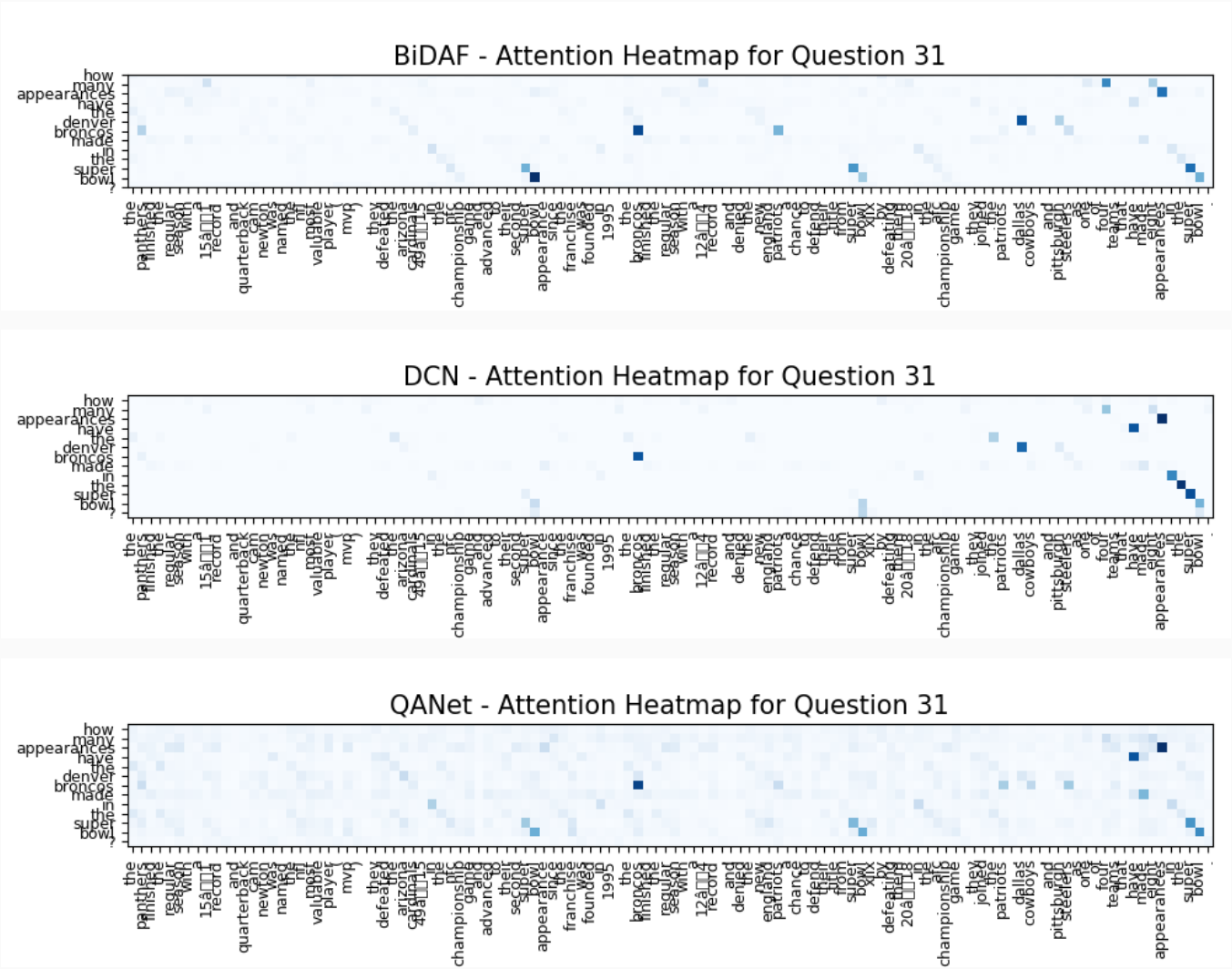
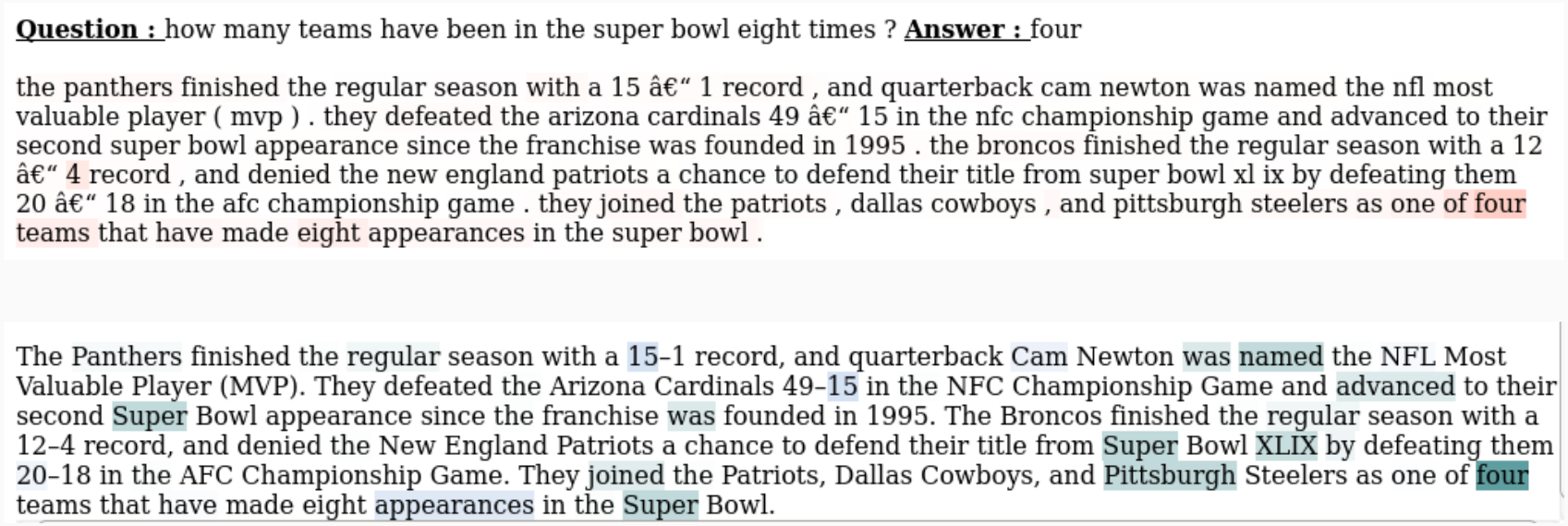


Table 10: Attention Mechanism Plot for BiDAF, DCN and QANet (top to bottom)

LIME versus Integrated Gradients



Visualization of importance scores -
Integrated Gradients (top) and LIME (bottom)

LIME highlights irrelevant words, maybe due to **lack of sensitivity**.

Other Preliminary Experiments

Pruning BERT's Layers

Layers pruned	%F1	%Drop in F1
None	88.73	-
11	88.66	0.07
10,11	87.81	0.85
9,10,11	86.58	1.23
8,9,10,11	86.4	0.18
7,8,9,10,11	85.15	1.25
6,7,8,9,10,11	83.75	1.4

Table 11: Pruned BERT models on SQuAD's dev-set

Conclusions :

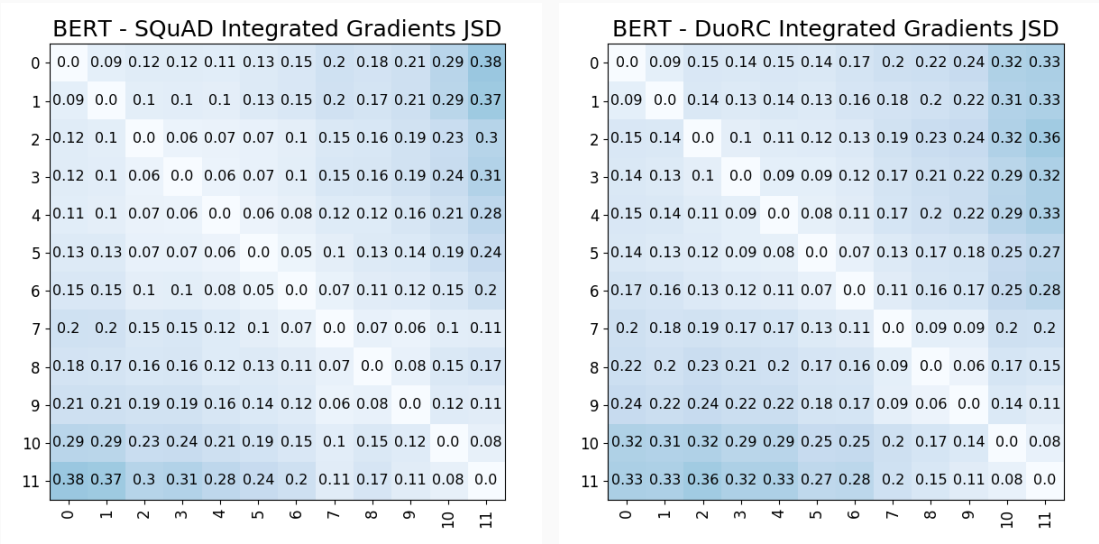
(i) Removing layers 11 and 8 cause almost no dip to the model's performance; perhaps they are redundant.

(ii) Removal of layers 10,9,7 cause a noticeable dip in performance; preliminarily, it can be said that they either contribute to the model's logic, or perform necessary mathematical perturbations in the model's high dimensions.

BERT on DuoRC

DuoRC explicitly requires the model to reason across multiple sentences.

BERT achieved an F1 of **54.9** on DuoRC, comparable to SOTA.



(a) **(b)**

Figure 14: JSD of integrated gradients scores, averaged across 500 datapoints for (a) BERT on SQuAD (b) BERT on DuoRC

Project Conclusion

Conclusion and Future Work

CONCLUSION

- ⇒ Analyzed four deep QA models using integrated gradients
- ⇒ Models give precise but incomplete rationale
- ⇒ Word embeddings tend to come closer as layers go deeper
- ⇒ BERT's initial layers focus on query-context interaction, later layers focus on answer prediction
- ⇒ Analysed qualitative examples for LIME and models' attention mechanisms

Conclusion and Future Work

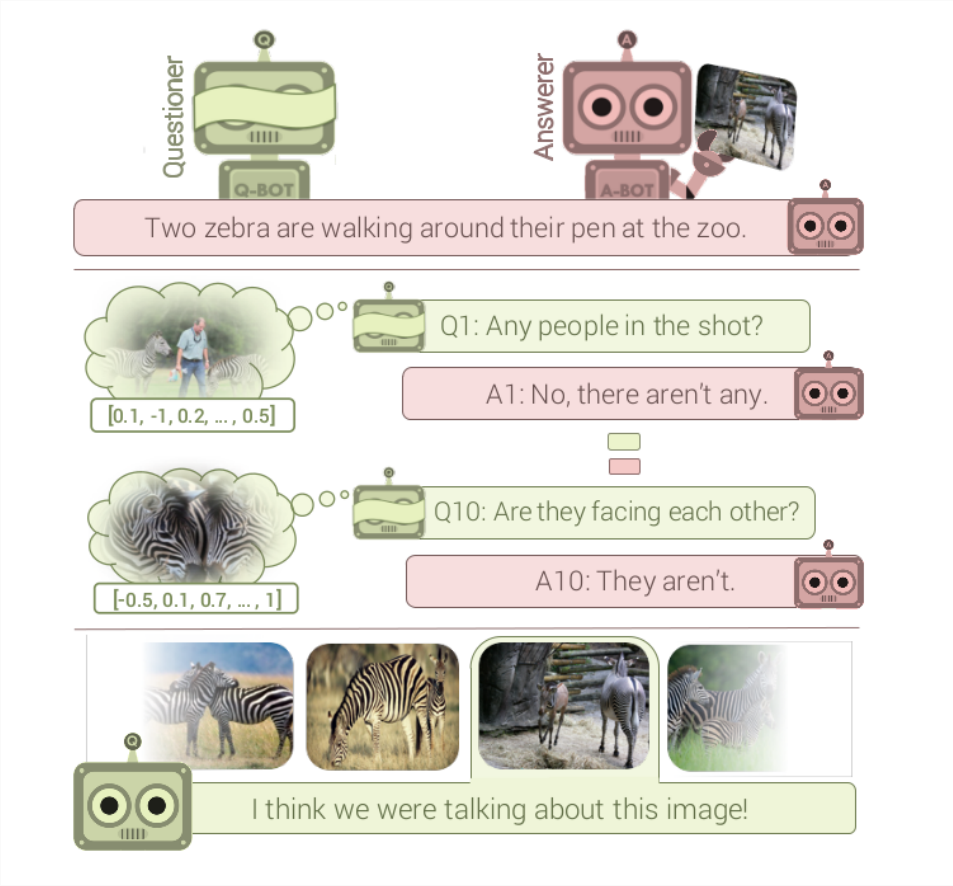
FUTURE WORK

$\Rightarrow$ Supplement this work, with analysis on a few more models/datasets

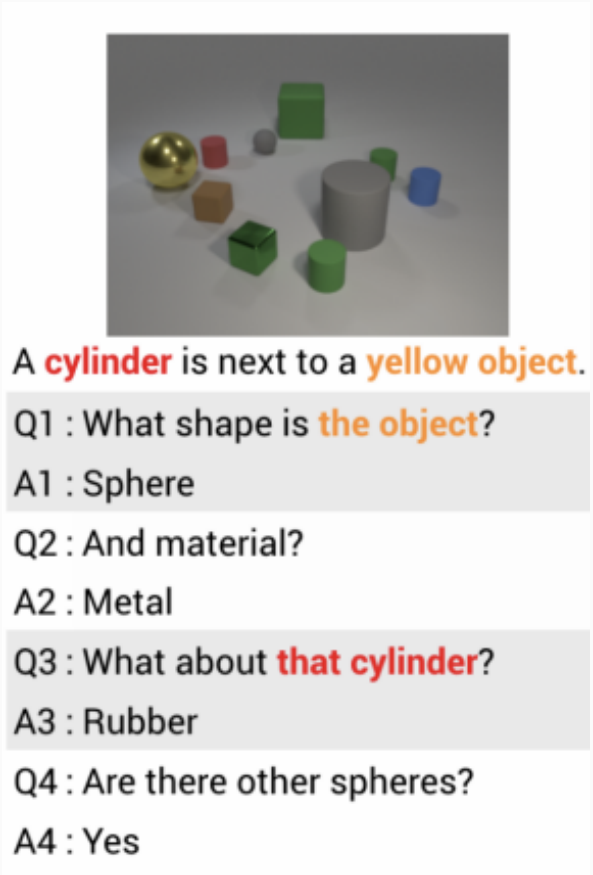
$\Rightarrow$ Further develop the preliminary pruning and DuoRC experiments

Dialog-Based Image Retrieval

Multimodal Conversation Datasets



VisDial Dataset [2]
Conversations about natural images from COCO



CLEVR Dialog Dataset [6]
Conversations about 3D objects from CLEVR with restricted attribute values

Traditional Multimodal Conversation Architecture

Most papers published on these datasets model **only** the answerer bot.

Effectively, the task reduces to answering a natural language question based on an image.

Traditional Multimodal Conversation Architecture

Most papers published on these datasets model **only** the answerer bot.

Effectively, the task reduces to answering a natural language question based on an image.

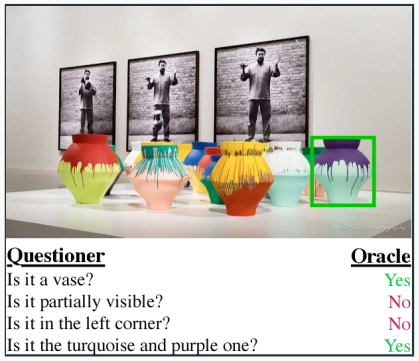
So far, there have been 2 kinds of approaches to solving this VQA task :

- Using vectorial representations
- Using neural symbolic-type reasoning

Modelling the Questioner Bot

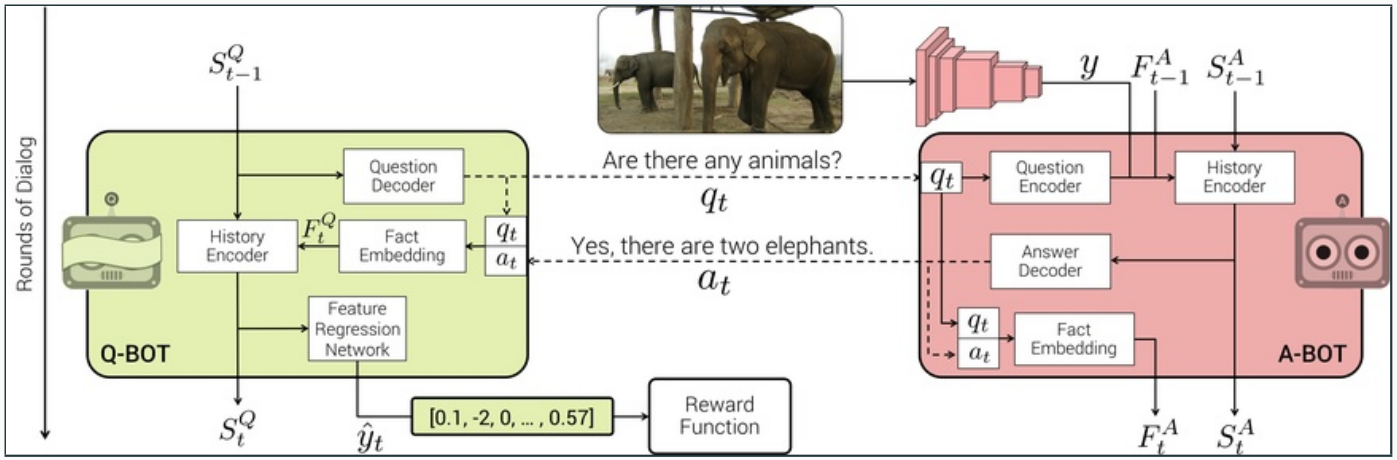
The few works which model the Q-Bot focus on two kinds of tasks :

- In an object detection setting : GuessWhat dataset [4]



Questioner	Oracle
Is it a vase?	Yes
Is it partially visible?	No
Is it in the left corner?	No
Is it the turquoise and purple one?	Yes

- In the image retrieval setting : Learning Cooperative Agents using Deep RL [3]



Use Cases

Why focus on modelling the Q-Bot?

- To use in domains where image search is required
  - Retail, travel, restaurants, social media
- Need a way to search interactively
- Interacting agent should ask relevant questions for better search

Project Statement

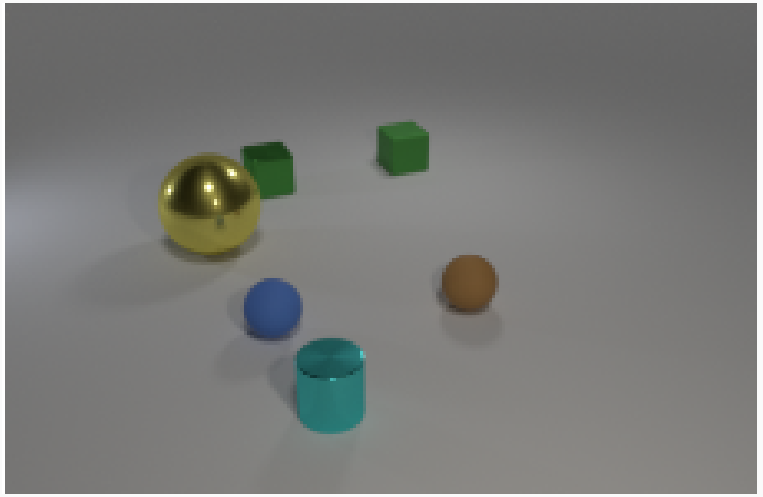
USING MULTIMODAL DIALOG FOR IMAGE RETRIEVAL

Broadly, the task is :

- $\Rightarrow$ Two way modelling of the Q-Bot and the A-Bot
- $\Rightarrow$ Image retrieval as a strategic search problem where the goal is reached through multiple iterations of the Q-Bot and A-Bot
- $\Rightarrow$ Modelling this strategic search through neural symbolic reasoning over scene graph representations of the images and the image catalog.

Image Retrieval Setting

1. A-Bot picks **image** from catalog.
2. A-Bot gives **caption** about it.
3. Q-Bot hallucinates image, asks **question**.
4. A-Bot gives **answer**.
5. Steps 3,4 repeat till Q-Bot **retrieves** image.



C	:	A **green block** is to the back of all objects.
Q1	:	If there is a object to the right of **it**, what size is it?
A1	:	Small
Q2	:	How about color?
A2	:	Brown
Q3	:	And material?
A3	:	Rubber
Q4	:	What about **that green object**?
A4	:	Rubber
Q5	:	How many other objects have the same color as **that small object**?
A5	:	0

Dataset Used

There are multiple options for datasets in this setting, such as VisDial, CLEVR Dialog, Fashion IQ, etc.

We have picked **CLEVR Dialog** as our initial dataset, since :

- Images and dialogs have restricted domains,
- Dataset has small number of object types, attribute values and relations.

CLEVR Dialog Advantages

These properties provide a simplistic setting in which we can thoroughly analyse the working(abilities, biases and limitations) of the model.

CLEVR Dialog Advantages

These properties provide a simplistic setting in which we can thoroughly analyse the working(abilities, biases and limitations) of the model.

We can use this as an **ideal setting** with perfect assumptions on which we tune the model structure, and then we extend the same to a more complex setting.

CLEVR Dialog Advantages

These properties provide a simplistic setting in which we can thoroughly analyse the working(abilities, biases and limitations) of the model.

We can use this as an **ideal setting** with perfect assumptions on which we tune the model structure, and then we extend the same to a more complex setting.

We are structuring our model such that it is generalizable to any setting so that we can eventually extend it to other, more complex datasets.

Our Proposed Pipeline

Usage of Scene Graphs

Previous works on image retrieval using dialog are modelled directly in the text and image vector spaces.

$\Rightarrow$ The Q-Bot encodes the conversation so far into a single context vector, which is projected into the image vector space.

$\Rightarrow$ The A-Bot combines the encoding of the question and the image to generate the answer.

While this does give good results, it faces the problem of **lack on interpretability** on how the Q-Bot and A-Bot are actually working.

Usage of Scene Graphs

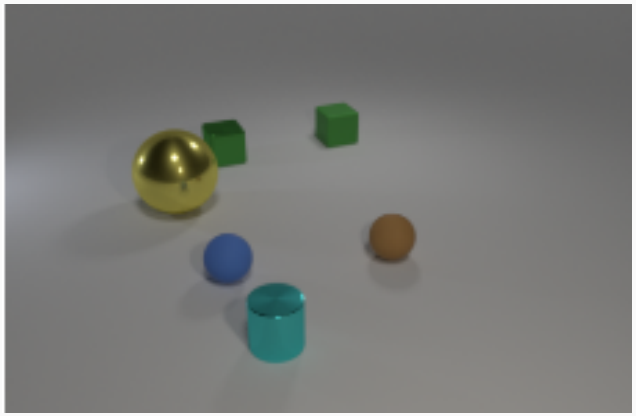
1. Q-Bot needs to **use catalog images** to ask relevant questions.

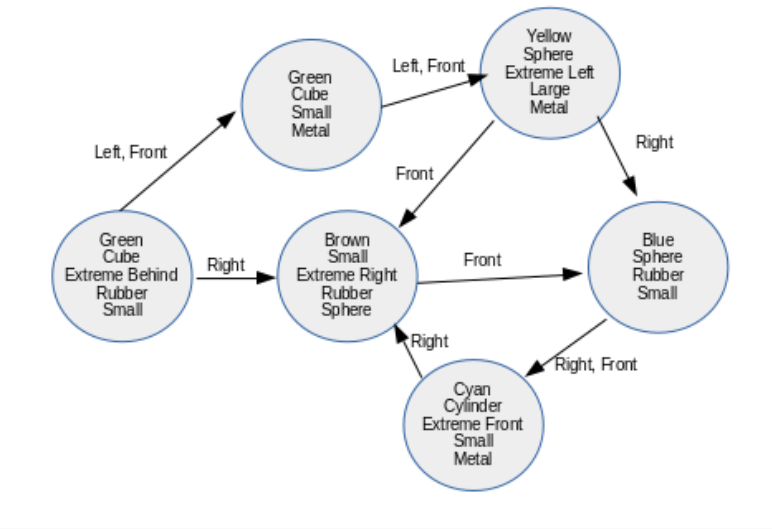
2. Q-Bot also needs to use the dialog to **logically update** its hallucination of the image.

3. A-Bot needs to answer questions in a **helpful manner**, instead of repetitive or safe responses.

4. Q-Bot needs a **more refined** method to map its final hallucination to an image in the catalog than just euclidean distance.

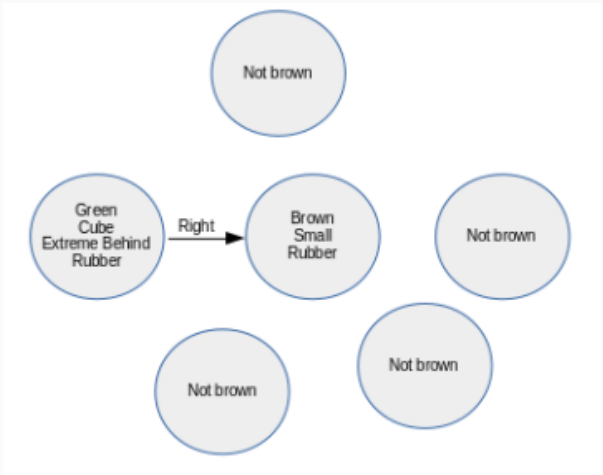
Scene Graph of Image and Dialog





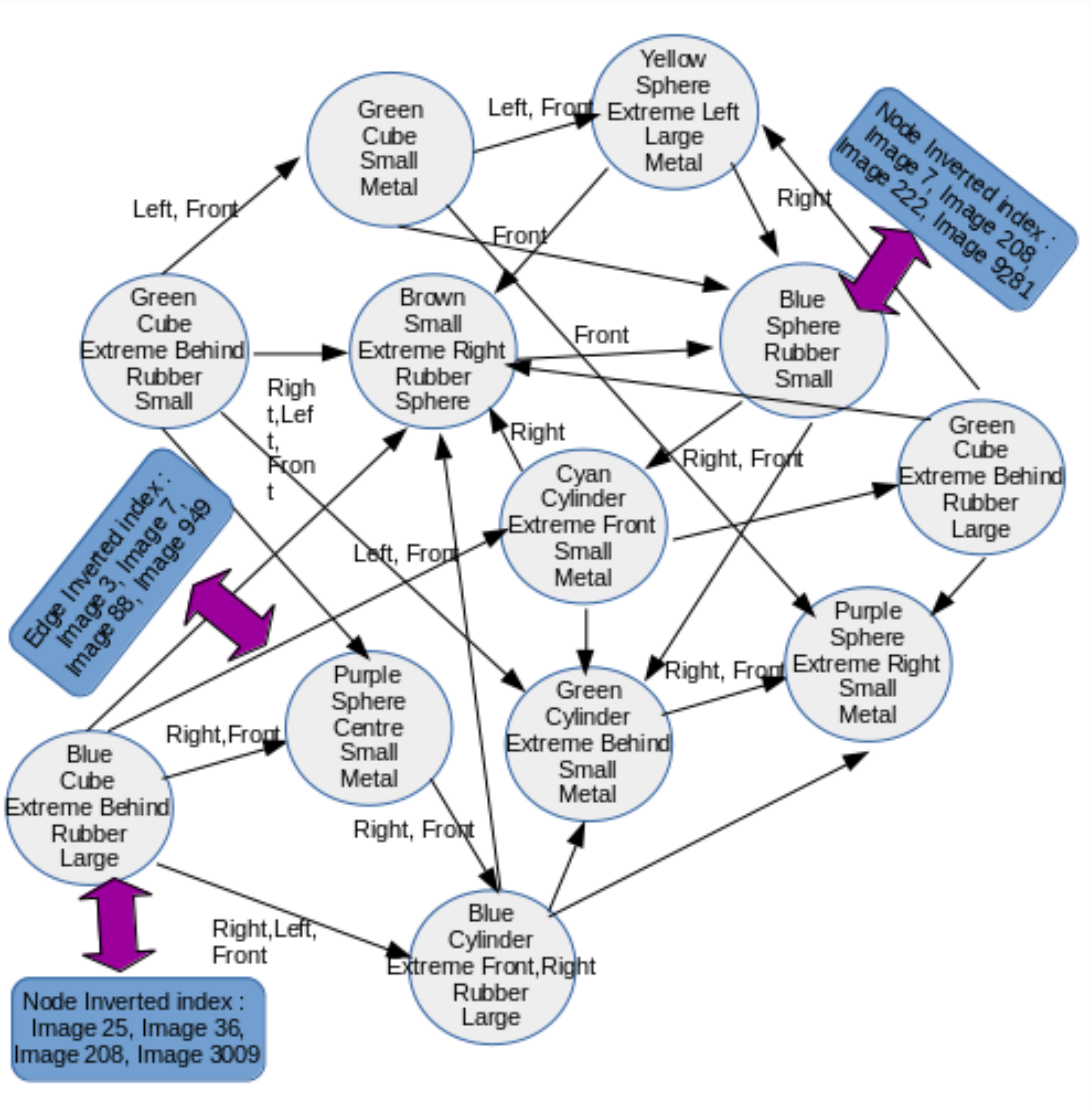
The image's full scene graph

C	:	A **green block** is to the back of all objects.
Q1	:	If there is a object to the right of **it**, what size is it?
A1	:	Small
Q2	:	How about color?
A2	:	Brown
Q3	:	And material?
A3	:	Rubber
Q4	:	What about **that green object**?
A4	:	Rubber
Q5	:	How many other objects have the same color as **that small object**?
A5	:	0



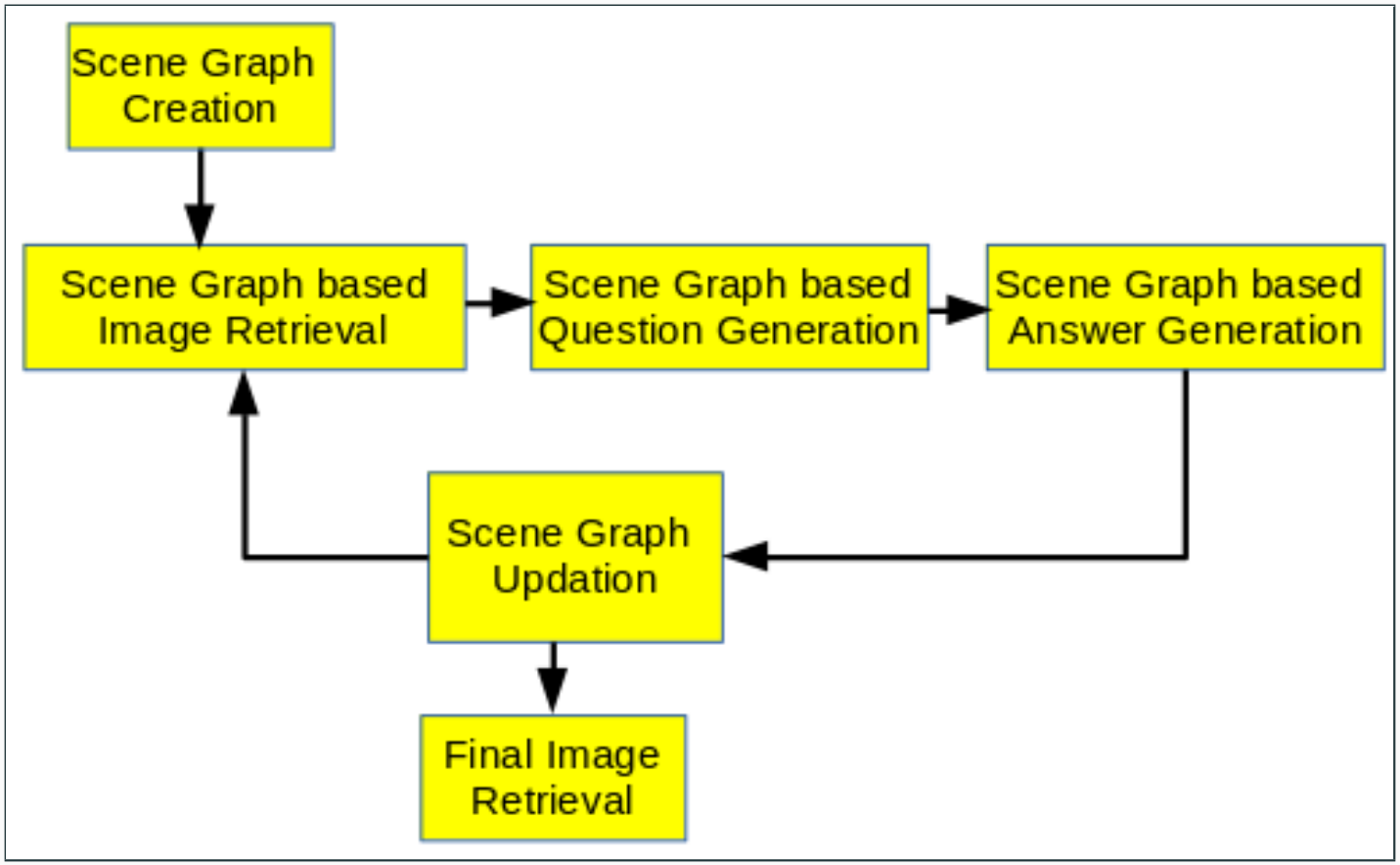
Partial scene graph from the dialog.
Called the **query scene graph**.

Catalog Scene Graph



Global Catalog Graph over all images in set.

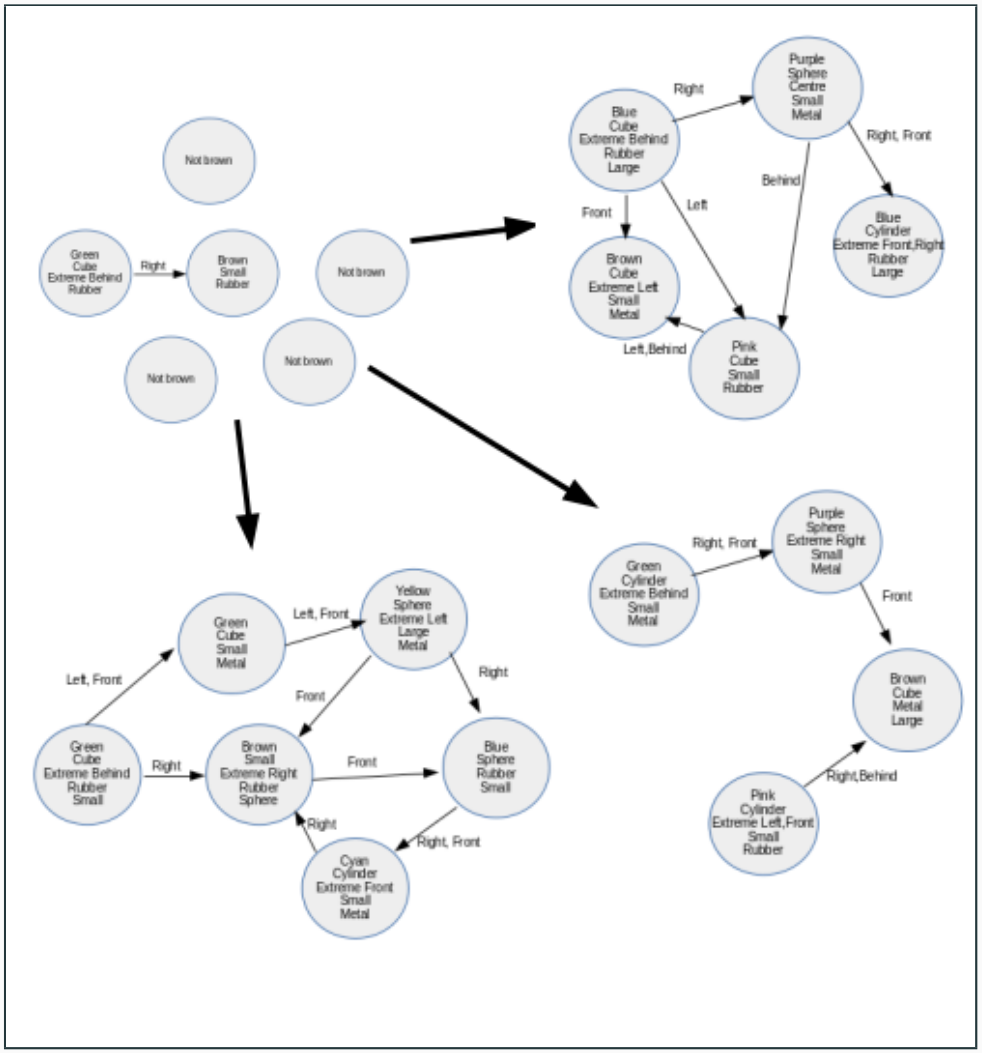
Proposed Pipeline



Our proposed pipeline

One Shot IR: Individual and Catalog Scene Graphs

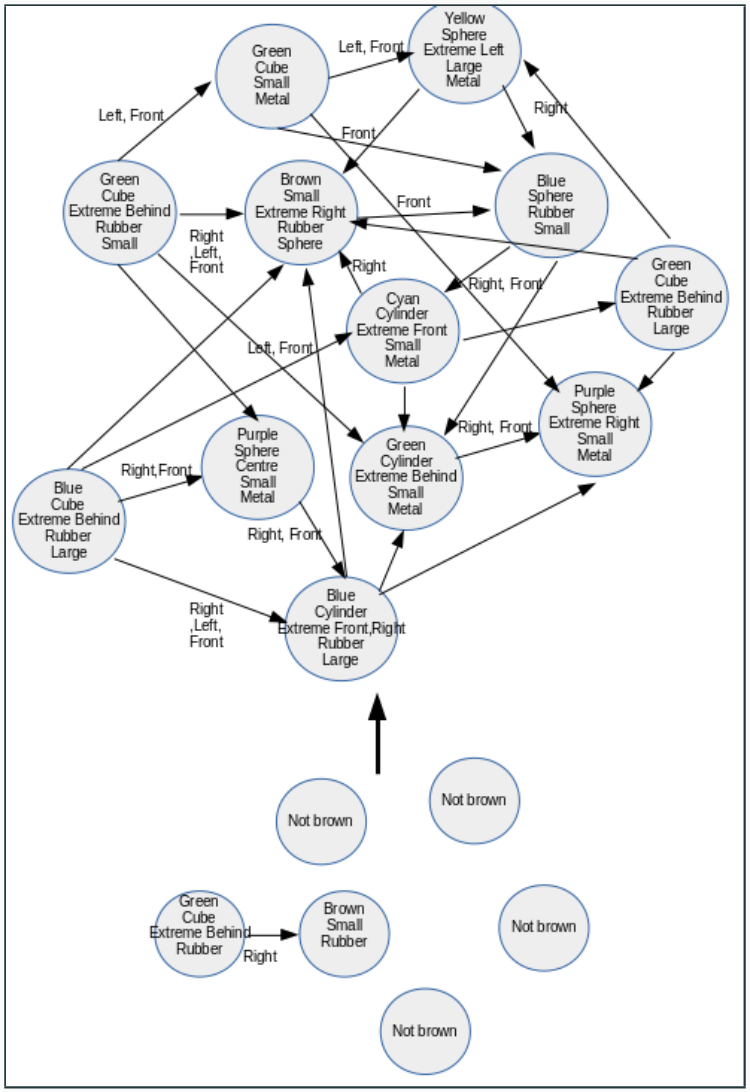
Based on the query scene graph, there are two possible methods to retrieve the image from the set of images.



(1) Compare query scene graph with all gold image graphs and retrieve most similar graph.

One Shot IR: Individual and Catalog Scene Graphs

Based on the query scene graph, there are two possible methods to retrieve the image from the set of images.



(2) Compare query scene graph with catalog graph and retrieve most similar subgraph.

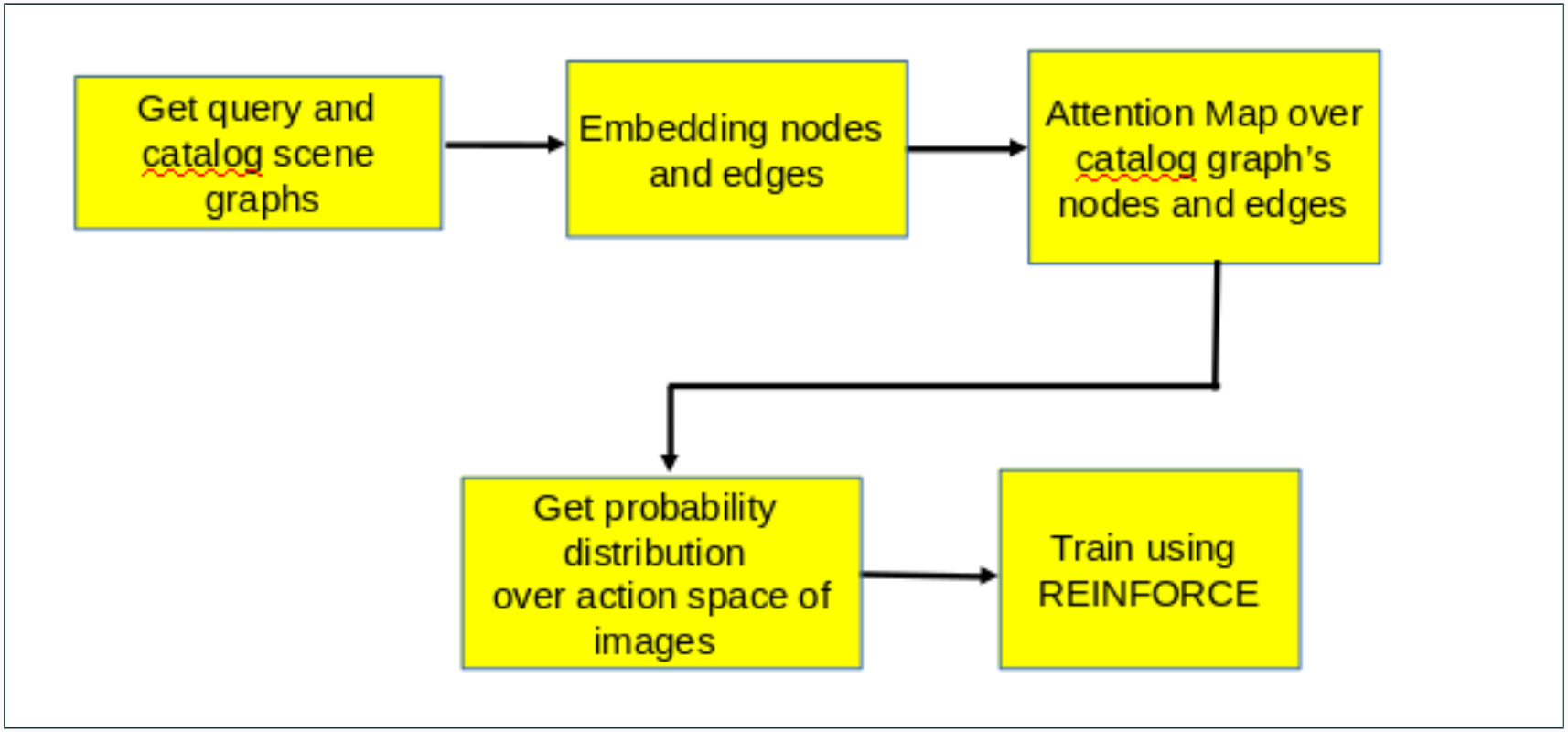
One Shot IR: Using the Catalog Scene Graph

Advantages of using a global catalog graph :

$\Rightarrow$ Matching a partial graph to subgraphs of one huge graph is more feasible than matching it to multiple (relatively) big graphs.

$\Rightarrow$ It is easier to get information from a single graph rather than aggregating it from multiple graphs, to ask the next question.

The Image Retrieval Module



The Image Retrieval Module

Embedding the nodes and edges

EMBEDDING THE NODES

Each node is represented as a list of 6 fixed attributes : [shape, colour, size, material, uniqueness, extremeness] where each of these can take on a fixed number of values only.

$$\boxed{N_{emb} = \text{matrix of disentangled attributes}}$$

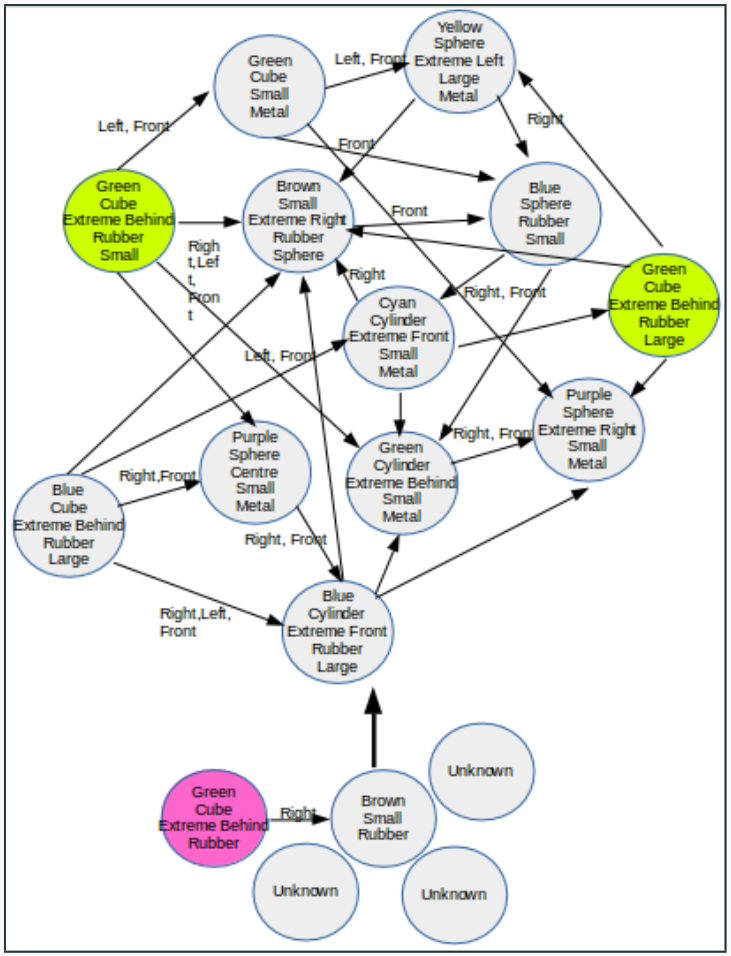
EMBEDDING THE EDGES

Each edge is represented as either (head node, relation type) or (head node, relation type, tail node).

$$\boxed{E_{emb} = \text{matrix of head node, relation (and tail node)}}$$

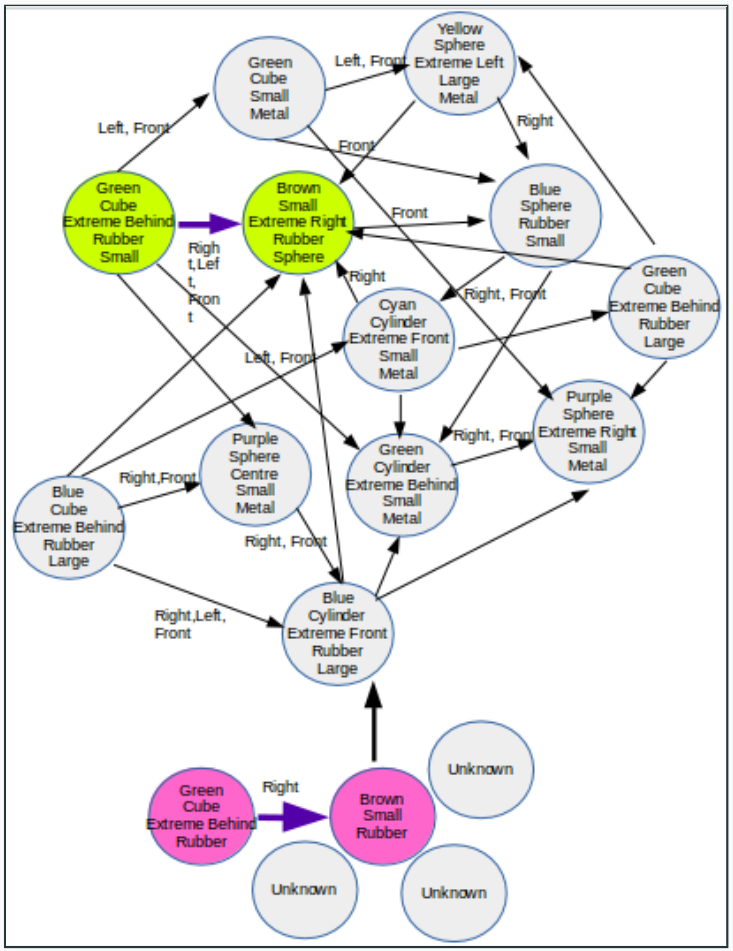
Currently using GloVe embeddings here, have to move to a multimodal setting.

Attention Map over nodes



An attention map is calculated over the catalog graph's nodes for each node in the query graph using the **frobenius norm** between the embeddings.$(\frac{1}{1+d})$

Attention Map over triples



Now, an attention map is calculated between each head-edge-tail triples of the query and catalog graph, using the **frobenius norm** between the head+tail node attributes and the relation type.($\frac{1}{1+d}$).

Action space for Image Retrieval

The **action space** is defined as all the **images** which the Q-Bot has access to.

$$P(\text{action=image}) = \prod_{\text{query nodes}} max(\text{score catalog nodes/edges} \in \text{image})$$

In this way, we score each image according to its relevance to each query node.
This ensures that the best image(s) according to **all** the query nodes are selected.

Training using REINFORCE

Training is done in a RL setting with the images as the actions and the policy as obtained above.

$$\theta_{t+1} = \theta_t + \alpha \sum_{images} P(\text{image})[R(\text{image}) - B] \log(P(\text{image}))$$

Weights are updated by increasing the REINFORCE reward, which is reducing euclidean distance between predicted and gold image. Baseline is a running average over rewards.

One Shot IR - Results

We tested our retrieval module on complete as well as simulated partial scene graphs from the CLEVR dataset.

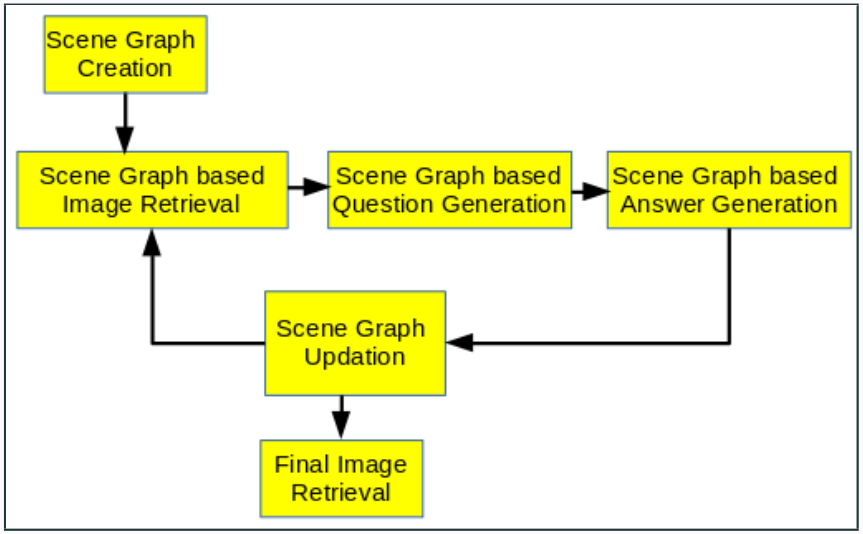
Setting	On average	Accuracy
With CLEVR gold scene graphs	6 nodes, 6 attributes, 20 edges	$\simeq 99.9\%$
Simulated partial scene graphs 0.8	5 nodes, 5 attributes, 18 edges	$\simeq 89\%$
Simulated partial scene graphs 0.7	4 nodes, 4 attributes, 15 edges	$\simeq 75\%$

Note that these numbers are for exact match with the target image.

Moving On to Full Pipeline

With the one-shot retrieval pipeline as the backbone, an iterative retrieval process is built for the Q-Bot and A-Bot.

Full Iterative Dialog-Based Retrieval



The training for these modules over the full dialog :

- At the 0^{th} round, scene graph based image retrieval is done
- At the i^{th} round :
  - Q-Bot samples from action space, generates question
  - A-Bot answers the question
  - Q-Bot updates query graph based on the answer
  - Scene graph based image retrieval is done
  - Modules are trained end-to-end based on retrieval reward

Action Space for Q-Bot

The question asked/action taken by Q-Bot must satisfy two criteria :

1. it must add new information to the query graph (valuable)
2. it must be the question that adds the most information to the query graph (non-redundant)

Action Space for Q-Bot

The question asked/action taken by Q-Bot must satisfy two criteria :

1. it must add new information to the query graph (valuable)
2. it must be the question that adds the most information to the query graph (non-redundant)

Two types of actions are made available to the QBot in the full pipeline :

1. **node attribute** : pick a node and ask for an unknown attribute value
2. **edge-value** : pick a node and ask about the presence/absence of its edges

Action Space for Q-Bot

Both these actions are termed as *categorical* actions, and are considered as representative of multiple *boolean* actions.

What is the color of this node? is a superset of **Is the color blue? Is the color red? etc.**

What is to the left of this node? is a superset of **Is the left edge present? Is the left edge absent?**

Embedding and Attention over Boolean Actions

EMBEDDING BOOLEAN ACTIONS

For each query node(QN), multiple attribute and edge boolean actions are created.

N_{emb} = QN's attr.s, one unknown attr. replaced with a wildcard value

E_{emb} = QN's edges, one unknown edge replaced with *edge-value* or *none*

ATTENTION MAP FOR BOOLEAN ACTIONS

Attention map over catalog nodes/edges created, treating boolean actions as nodes/edges.

Probability of Actions

ACTION SCORES

Each boolean action gets a score :

$$\boxed{action\_score = attn(action) \times attn(query\ \ node/edge)}$$

PROBABILITY/SHATTER SCORE

Each node/edge in the catalog graph is assigned a tally of how many images they are a part of.

$$\boxed{shatter\_score = \sum_{catalog} action\_score \times tally}$$

The shatter score is normalized to get a probability distribution. It is a measure of how well the each boolean action splits the image space.

Entropy of Actions

ENTROPY OF ACTIONS

Entropy is calculated across boolean actions to get a score for each *categorical action*.

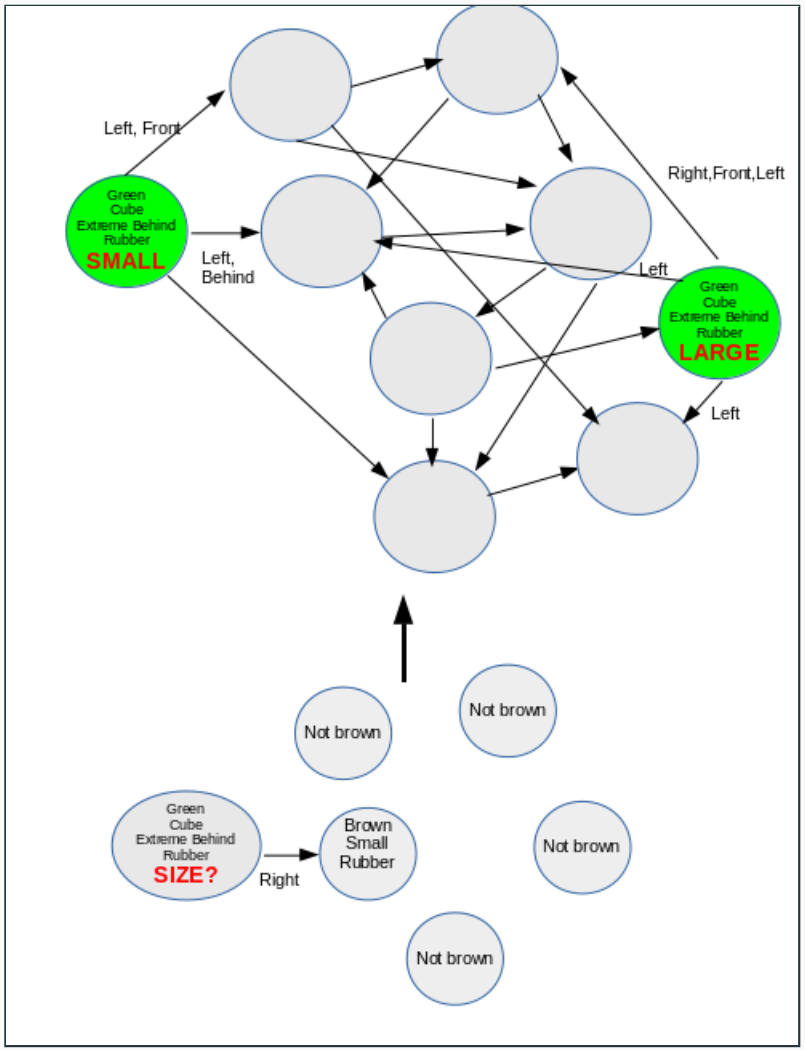
$$\boxed{\textit{categorical\_score} = \textit{entropy}(\textit{boolean})}$$

ACTION PICKED

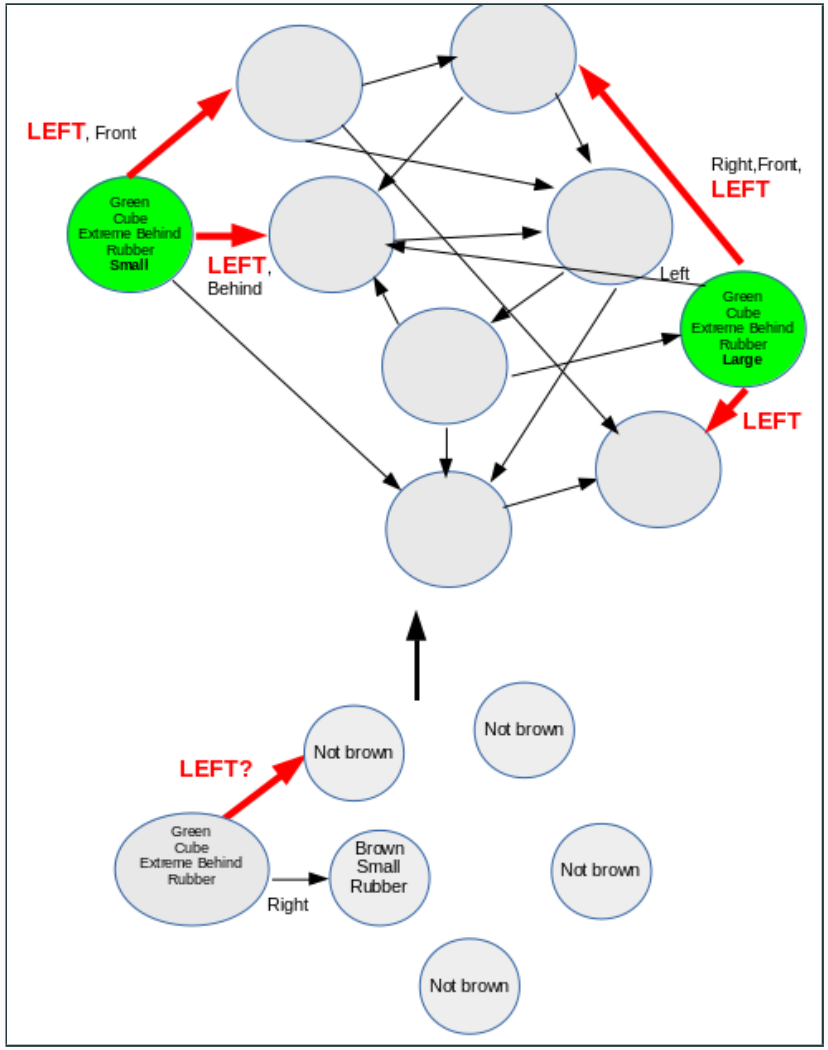
QBot selects the categorical action which has the *minimum* entropy, helping to narrow down the image space as much as possible at each round of conversation.

$$\boxed{\text{action picked} \leftarrow \textit{argmin}(\textit{categorical\_score})}$$

Joint Functioning of QBot and ABot



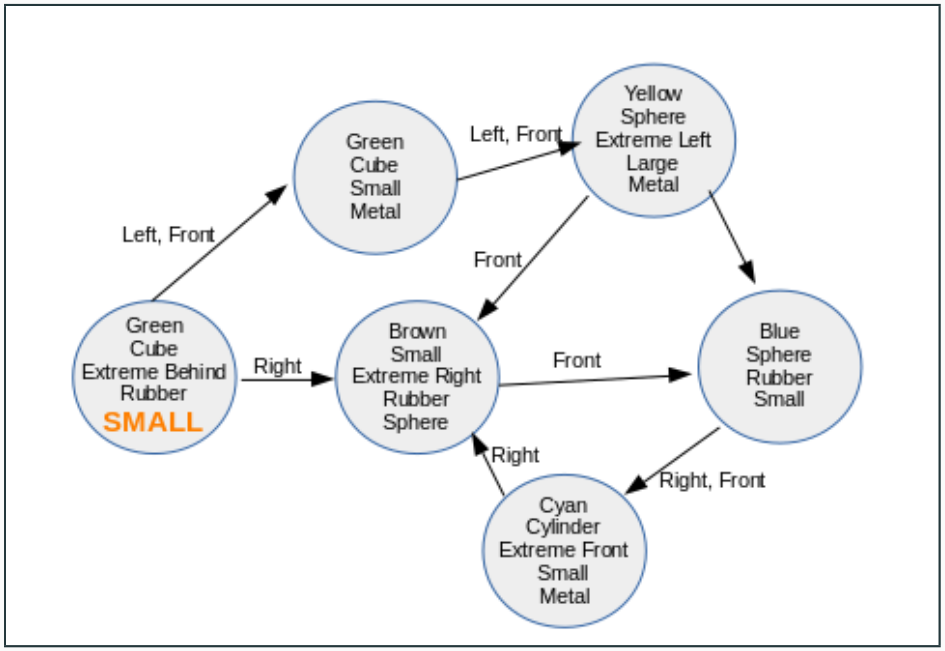
Question on node attributes

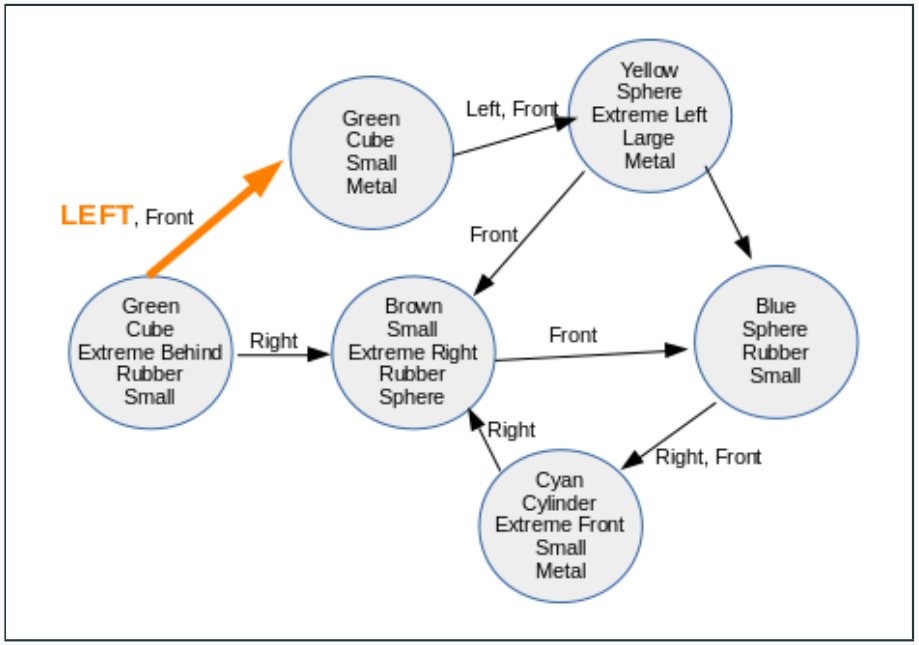


Question on edge presence

The QBot picks an action and sends it to the ABot.

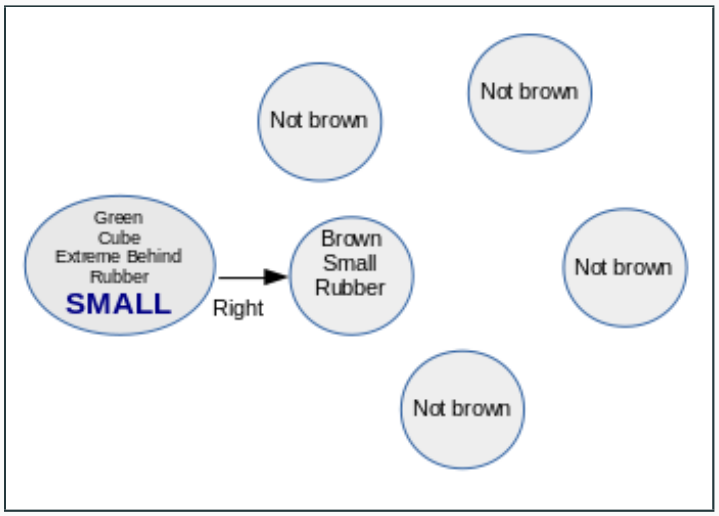
Joint Functioning of QBot and ABot

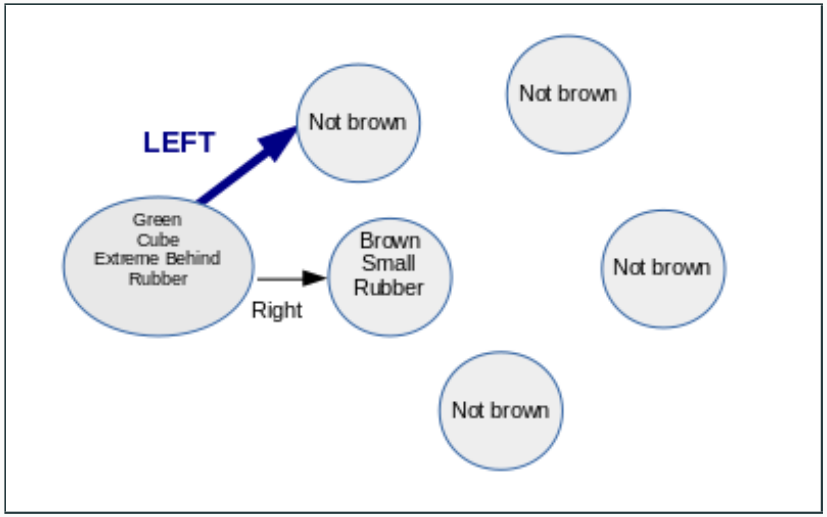




The ABot compares the question and node asked to its gold image scene graph and returns an appropriate answer.

Joint Functioning of QBot and ABot





The QBot then uses this answer and updates its query scene graph accordingly.

Experiments - Iterative Image Retrieval

The full iterative retrieval model is tested on the validation set of CLEVR Dialog dataset.

Question Types	5 rounds	10 rounds	20 rounds	40 rounds
Node Attributes	27.4	61.6	83.06	-
Node Attributes, Edge Type	8.12	15.3	48.89	88.32

Table 12: Image retrieval accuracy (%) after iterative rounds of QBot-ABot conversation, using dialog gold query graphs as input and node-attribute and and edge-type based questions.

(Note that these numbers are for exact match of retrieved image with the target image)

Project Conclusion

Conclusion and Future Work

CONCLUSION

- ⇒ Implemented a full iterative neural-symbolic model for dialog-based image retrieval
- ⇒ Observed competitive results on CLEVR and CLEVR Dialog

Conclusion and Future Work

CONCLUSION

⇒ Implemented a full iterative neural-symbolic model for dialog-based image retrieval

⇒ Observed competitive results on CLEVR and CLEVR Dialog

FUTURE WORK

⇒ Use generated probabilistic SGs over exact gold SGs

⇒ Implement higher level questions(triplets, higher-order subgraphs)

⇒ To extend this neural-symbolic model to a natural setting

Conclusion

Conclusion

Deep Learning models are now being used in almost all aspects of life. Now more than ever, it is important to ensure that these models are interpretable and can be trusted.

In this project, I've worked on analyzing interpretability of four existing deep RCQA systems and on implementing an explainable neural-symbolic image retrieval system. The 2 projects gave interesting insights into how interpretable existing systems are, and how to develop/analyze new models to ensure that they are transparent.

References i


K. Clark, U. Khandelwal, O. Levy, and C. D. Manning.
What does BERT look at? an analysis of bert's attention.
CoRR, abs/1906.04341, 2019.


A. Das, S. Kottur, K. Gupta, A. Singh, D. Yadav, J. M. Moura, D. Parikh, and D. Batra.
Visual dialog.
In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 326–335, 2017.


A. Das, S. Kottur, J. M. Moura, S. Lee, and D. Batra.
Learning cooperative visual dialog agents with deep reinforcement learning.
In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2951–2960, 2017.

References ii


H. De Vries, F. Strub, S. Chandar, O. Pietquin, H. Larochelle, and A. Courville.
Guesswhat?! visual object discovery through multi-modal dialogue.
In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5503–5512, 2017.


R. Jia and P. Liang.
Adversarial examples for evaluating reading comprehension systems.
CoRR, abs/1707.07328, 2017.


S. Kottur, J. M. Moura, D. Parikh, D. Batra, and M. Rohrbach.
Clevr-dialog: A diagnostic dataset for multi-round reasoning in visual dialog.
arXiv preprint arXiv:1903.03166, 2019.

References iii


T. Lei, R. Barzilay, and T. Jaakkola.
Rationalizing neural predictions.
arXiv preprint arXiv:1606.04155, 2016.


M. E. Peters, M. Neumann, L. Zettlemoyer, and W.-t. Yih.
Dissecting contextual word embeddings: Architecture and representation.
arXiv preprint arXiv:1808.08949, 2018.


S. Serrano and N. A. Smith.
Is attention interpretable?
arXiv preprint arXiv:1906.03731, 2019.

References iv


A. Sharma, P. Talukdar, et al.
Towards understanding the geometry of knowledge graph embeddings.
In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 122–131, 2018.


C. Si, S. Wang, M.-Y. Kan, and J. Jiang.
What does bert learn from multiple-choice reading comprehension datasets?
arXiv preprint arXiv:1910.12391, 2019.


I. Tenney, D. Das, and E. Pavlick.
Bert rediscovers the classical nlp pipeline.
arXiv preprint arXiv:1905.05950, 2019.

THANK YOU!